ALLEN-BRADLEY
A ROCKWELL INTERNATIONAL COMPANY

User's Manual

Bulletin 5370
CVIM™
MATH-PAK™ Option
(Cat. No. 5370-MPK)

40062-194-01(A)

Catalog No. 5370-ND005 (Series A)

Important User Information

Solid-state equipment has operational characteristics differing from those of electromechanical equipment. "Application Considerations for Solid-State Controls" (Publication SGI-1.1) describes some important differences between solid-state equipment and hard wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid-state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Allen-Bradley Company be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Allen-Bradley Company cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Allen-Bradley Company with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of the Allen-Bradley Company is prohibited.

© 1991 Allen-Bradley Company

Table of Contents

Title	Page
CVIM MATH-PAK: Installation and Configuration	
MATH-PAK User Manual: Objectives	1
MATH-PAK Option: Definition	1
User Experience	1
MATH-PAK Option: Overview	2
Installing MATH-PAK Option	4
MATH-PAK Option: Menu/Display Changes	6
Gages and Windows: Range/Reference Popup Menu	6
Misc Popup Menu	7
Exit Popup Menu	9
Range Fail Menu Box	9
Disp. Formulas Menu Box	9
Results Page Display	10
Stat 1 Page Display	11
I/O Page Display	12
All Tools Display	12
Configuring MATH-PAK Formulas	13
Definitions: MATH-PAK Terms and Concepts	16
Definitions: Formula Entry Key Functions	18
Digit Keys: 0 to 9	18
Operand Keys	19
Operator Keys	21
Control Keys	22
Using Control Keys	23
Del (Delete) Key	23
Clr (Clear) Key	24
Ret (Return) Key	25
Esc (Escape) Key	26
Cursor Keys: (←) and (→)	26
Keyboard Select Keys (↑) and (↓)	27
Using Operator and Operand Keys	28
ABS Key	28
ATN Key	30
AVG Key	35
COS Key	37
DST Key	39
MAX, MIN Keys	41
MED Key	43
MOD Key	45
SIN Key	47
SQR Key	48
SQ Key	50
TAN Key	51
VAR Key	53
Arithmetic Operator Keys: +,–,*,/	55

Title	Page
Parenthesis Keys: ( )	55
Modulo Operator Key: %	56
Unary Operator Key: ±	56
Comma Key: ,	56
Logic Operator Keys	56
Setting Formula Name	59
Enabling Formula	60
Assigning Range Limits and Outputs	62
Results Data: Size and Rounding Constraints	68
MATH-PAK Application Examples	69

List of Figures

Figure	Title	Page
1.1	CVIM System *Without* MATH-PAK Formulas	2
1.2	CVIM System *With* MATH-PAK Formulas	3

	Title	Page
Appendix A	**CVIM Communications With MATH-PAK**	
	Objectives	A-1
	CVIM With MATH-PAK Communications	A-1
	Location of Formula Results	A-1
	Data Formats For Formula Results	A-6
	Obtaining Formula Results and Statistics	A-6
	Statistics Block	A-6
	Configuration Blocks	A-7
	Fail/Warning Ranges: Formulas # 9-16, 25-32, 41-56 in Toolset 1	A-8
	Fail/Warning Ranges: Formulas # 1-8, 17-24, 33-40 in Toolset 1	A-9
	Fail/Warning Ranges: Formulas # 9-16, 25-32, 41-56 in Toolset 2	A-10
	Fail/Warning Ranges: Formulas # 1-8, 17-24, 33-40 in Toolset 2	A-11
	Discrete Bit Inputs	A-12

CVIM™ MATH-PAK™: Installation and Configuration

MATH-PAK User Manual: Objectives

The objectives of this manual are to introduce you to the MATH-PAK™ option for the Allen-Bradley 5370-CVIM (Configurable Vision Input Module) machine vision system and provide the information and procedures you need to install the MATH-PAK option in your CVIM system and configure the system for mathematics applications.

Here are the major sections in this manual:

MATH-PAK Option: Overview describes briefly how the CVIM system handles inspection results data when using MATH-PAK formulas.

Installing MATH-PAK Option shows you how to install the MATH-PAK option on your CVIM system.

MATH-PAK Option: Menu/Display Changes shows you how the MATH-PAK option affects the menus and display tables.

Configuring MATH-PAK Formulas describes the mathematical and logical operators available with the MATH-PAK option and shows you how to use them to construct MATH-PAK formulas.

MATH-PAK Application Examples shows you how MATH-PAK formulas can be used in various combinations to solve real-world application problems.

NOTE: It is beyond the scope of this manual to show you how to configure the CVIM system for *specific* MATH-PAK applications.

Appendix A provides information needed to access MATH-PAK results data using the CVIM communication ports. This information supplements the *CVIM Communications Manual*, Catalog No. 5370-ND002.

MATH-PAK Option: Definition

MATH-PAK is an option that enables your CVIM system to perform mathematical and/or logical operations, using formulas that you specify, on results data from any of CVIM's analysis tools. Your inspection application can use results data from these operations *instead* of, or in *addition* to, results data from individual analysis tools, according to your specific requirements.

User Experience

Before using MATH-PAK, you should be familiar with the CVIM system and its configuration procedures. The *CVIM User's Manual*, Catalog No. 5370-ND001, provides detailed information about the system's configuration and operation procedures. Chapters 9 and 10 are particularly relevant to MATH-PAK configuration and operation.

If your application uses the CVIM communications ports on the chassis backplane and/or front panel, refer also to the *CVIM Communications Manual*, Catalog No. 5370-ND002.

MATH-PAK Option: Overview

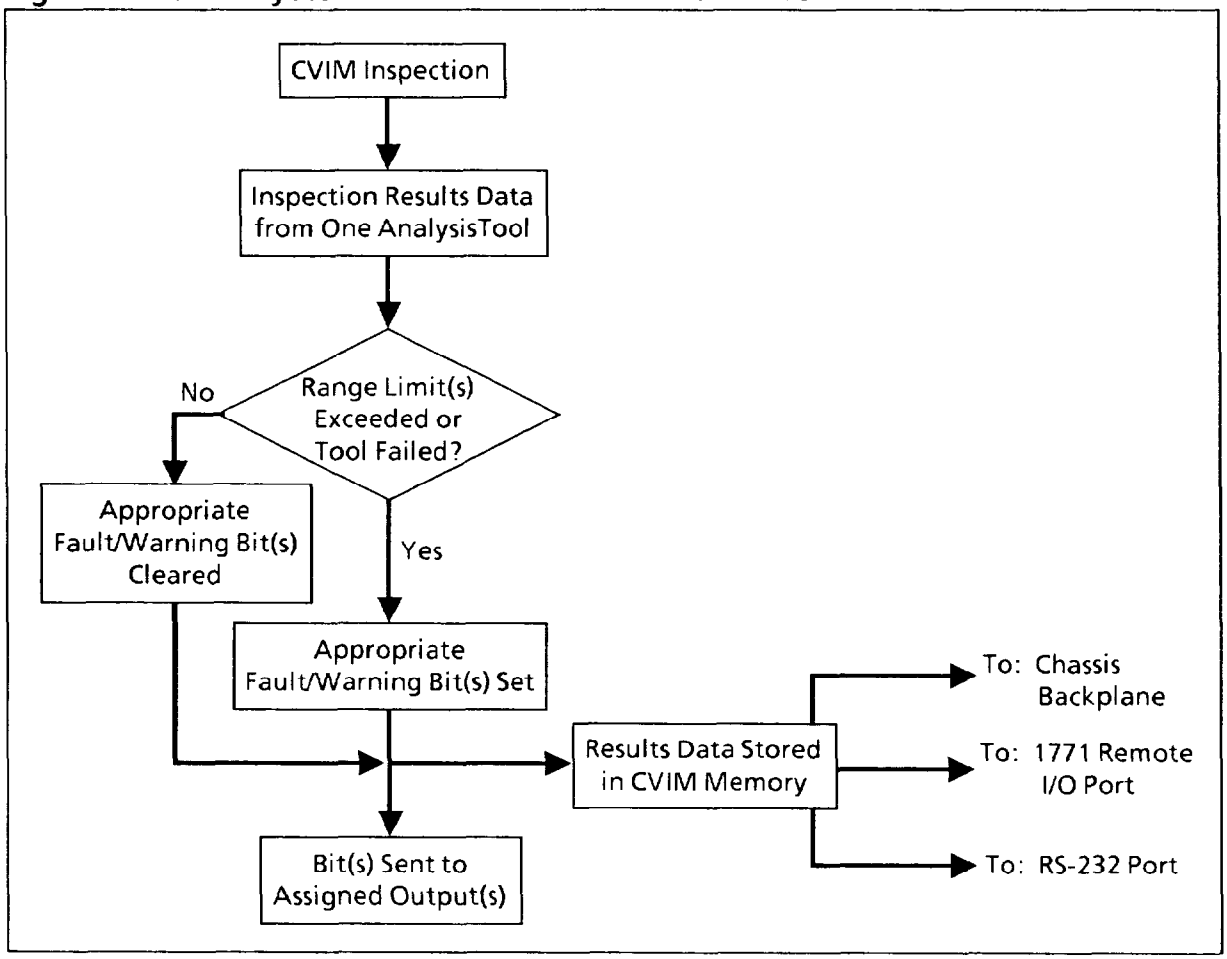
Following each inspection, the CVIM system stores in its memory the inspection results data from each enabled analysis tool (light probe, reference tools, gages, and windows). These data are available (until overwritten by the next inspection) to devices connected to the chassis backplane, the 1771 remote I/O port, the RS-232 port, and the discrete output port.

When operating *without* MATH-PAK formulas, the CVIM system compares the results data from each tool to the tool's specific warning and fault limits. Whenever the results data are above or below one of these limits, the system sets a discrete bit and routes it to the assigned output.

(Note that range limits do not apply to reference tools. These tools either pass or fail according to whether they find their edge or feature. If a reference tool fails, the CVIM system sets a discrete bit and routes it to the assigned output.)

Figure 1.1 shows, symbolically, how the CVIM system handles inspection results data when operating *without* using MATH-PAK formulas.

Figure 1.1 CVIM System *Without* MATH-PAK Formulas

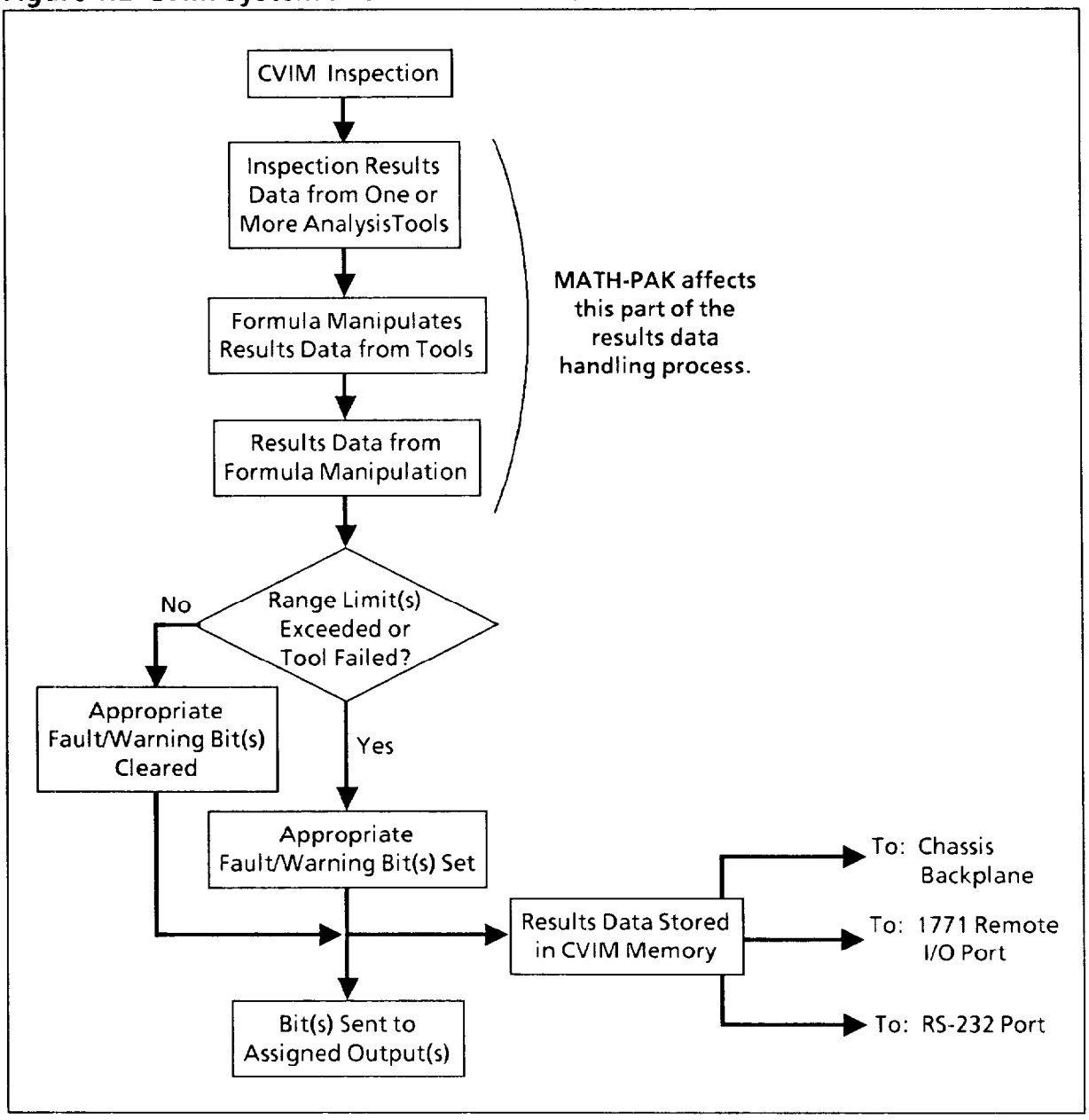


MATH-PAK Option: Overview(continued)

When the MATH-PAK option is installed in the CVIM system, you can configure formulas to perform *mathematical* and/or *logical* operations on the inspection results data from one or more analysis tools. The CVIM system processes the results data from the formula's operation in the same way that it handles results data from tools.

Figure 1.2 shows how the CVIM system handles inspection results data when the MATH-PAK option is installed and formulas are configured to manipulate results data from one or more tools.

Figure 1.2 CVIM System *With* MATH-PAK Formulas


CVIM Inspection
Inspection Results Data from One or More AnalysisTools
Formula Manipulates Results Data from Tools
Results Data from Formula Manipulation
MATH-PAK affects this part of the results data handling process.
Range Limit(s) Exceeded or Tool Failed?
No
Yes
Appropriate Fault/Warning Bit(s) Cleared
Appropriate Fault/Warning Bit(s) Set
Bit(s) Sent to Assigned Output(s)
Results Data Stored in CVIM Memory
To: Chassis Backplane
To: 1771 Remote I/O Port
To: RS-232 Port

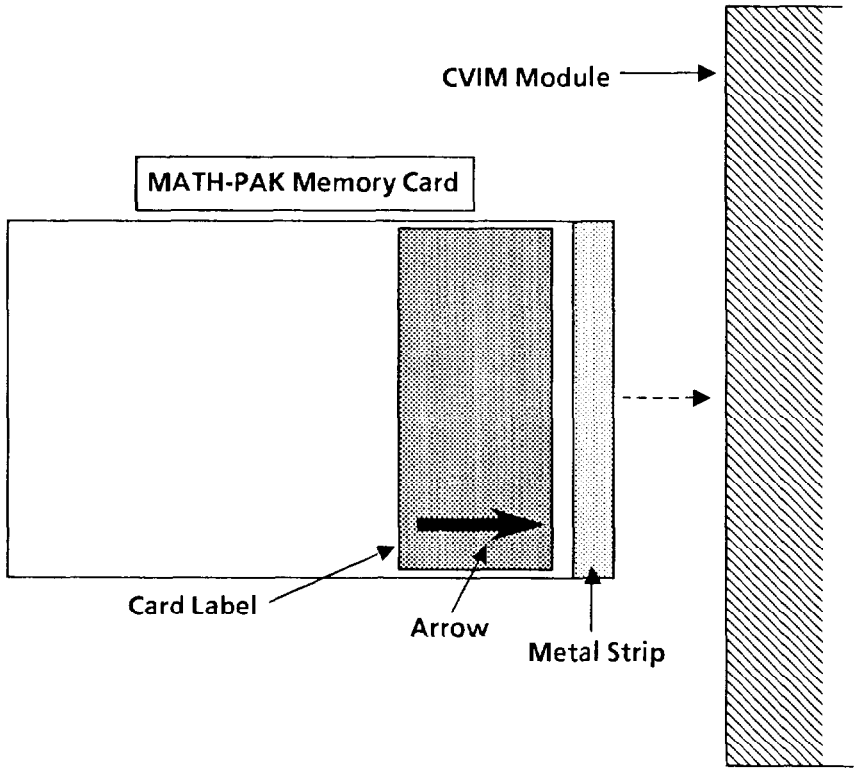
Installing MATH-PAK Option

The MATH-PAK option is provided on a memory card. You can install the option in your CVIM system by inserting the memory card into the Archive Memory slot on the CVIM module's front panel and applying power to the module.

NOTE: The CVIM system must have Revision C01 (or higher) firmware before the MATH-PAK option can be installed. (The firmware revision can be seen in the banner message on the monitor screen following powerup.)

The following steps assume that the CVIM system contains Revision C01 (or higher) firmware and is currently powered up. Use these steps to install the MATH-PAK option in your CVIM system.

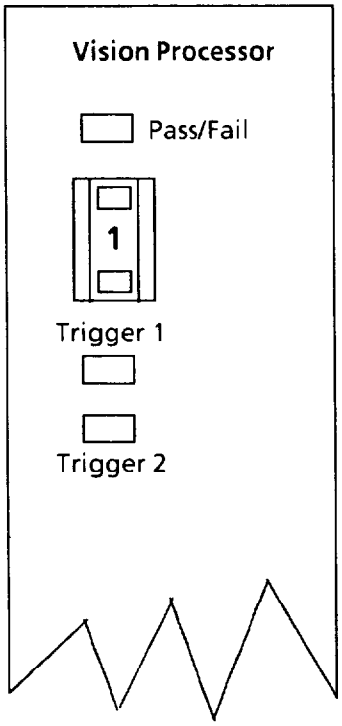
Your Action	Comments
Pull the DC/PWR LOCK *handle out as far as it will go.*	This removes DC power from the CVIM module.
Insert the memory card into the Archive Memory *slot on the CVIM module.*	The metal strip (and the arrow on the card label) must be facing right.



Push in the DC/PWR LOCK handle.	Be sure the lock handle is in as far as possible. This applies DC power to the CVIM module and begins loading the MATH-PAK software.

Installing MATH-PAK Option (continued)

Your Action	Comments
Observe the LEDs during software loading.	The top three LEDs on the CVIM module indicate the progress of the three-phase software loading process, as follows:



LEDs	MATH-PAK Loading Phase		
	1	2	3
Pass/Fail	Red-to-yellow-to-green.	Green	Red-to-yellow-to-green.
Trigger 1	Green	Green	Off
Trigger 2	Off	Green	Green

The MATH-PAK software loading process requires about 45 seconds, with each phase taking about 15 seconds. When the loading process is complete, the CVIM system begins its normal powerup sequence.

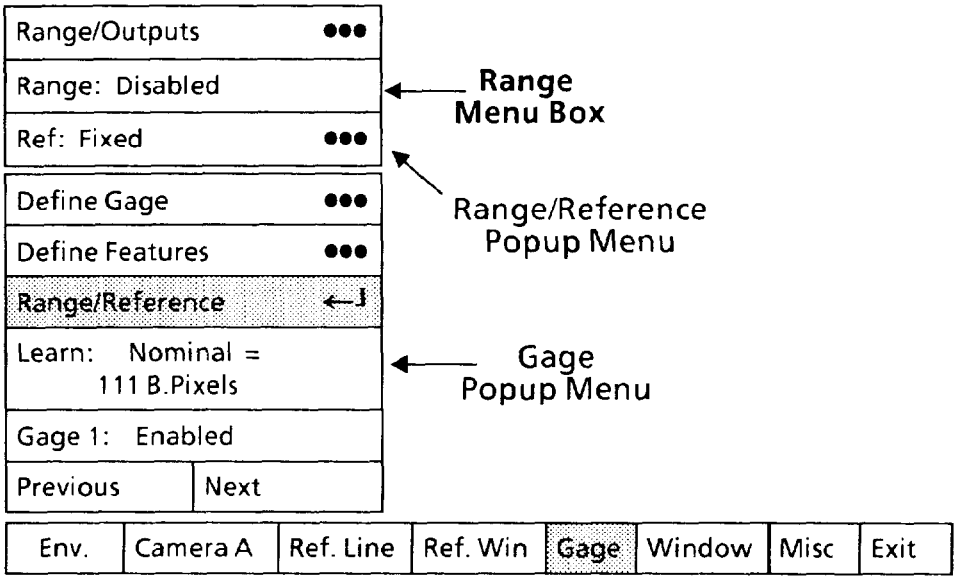
MATH-PAK Option: Menu/Display Changes

Since the MATH-PAK option adds functions to the basic CVIM system, installing the option changes the appearance of some of the menus and display tables in the user interface.

This section describes the changes resulting from the installation of the MATH-PAK option.

Gages and Windows: Range/Reference Popup Menu

When you pick the Range/Reference menu box in the Gage (or Window) popup menu, the Range/Reference popup menu appears with a new menu box, as follows:



The Range menu box indicates the status of range checking for the currently selected gage or window. When "Disabled" appears in the menu box, the CVIM system does *not* perform range checking for the tool. This is true in any of the following situations:

- When the currently selected tool is *disabled.*
- When the selected tool is *enabled,* but you intentionally select "Disabled" in the Range menu box.
- When an enabled formula preempts range checking for the selected tool. Thus, when formula #1 is *enabled,* range checking for window #1 is *disabled.*

"Enabled" appears in the Range menu box (and the CVIM system performs range checking for the currently selected tool) *only* when you complete the following steps:

- *Disable* the formula number that can preempt range checking for the currently selected tool. Thus, when formula #1 is *disabled,* range checking for window #1 can be *enabled.* (For more information about this subject, refer to the *Enabling Formula* heading in this manual.)
- *Enable* the selected tool.
- *Enable* range checking for the selected tool.

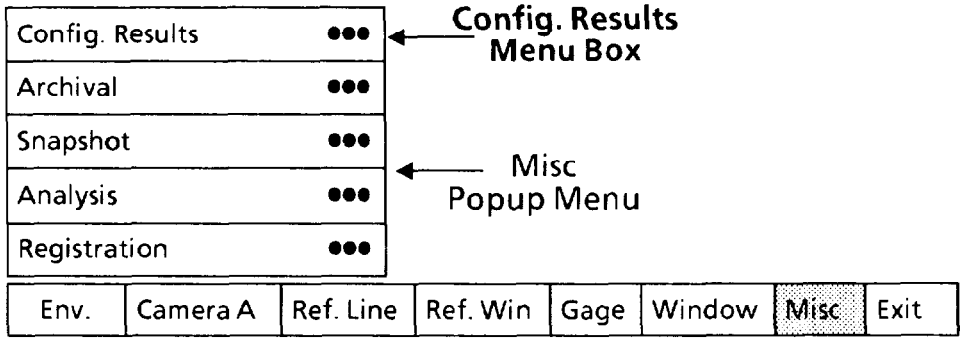
Gages and Windows: Range/Reference Popup Menu (continued)

When either of the first two conditions is *not* satisfied, "Disabled" appears in the Range menu box in **black type**, indicating that you cannot toggle the menu box to "Enabled." Thus, you *cannot* enable range checking for the selected tool.

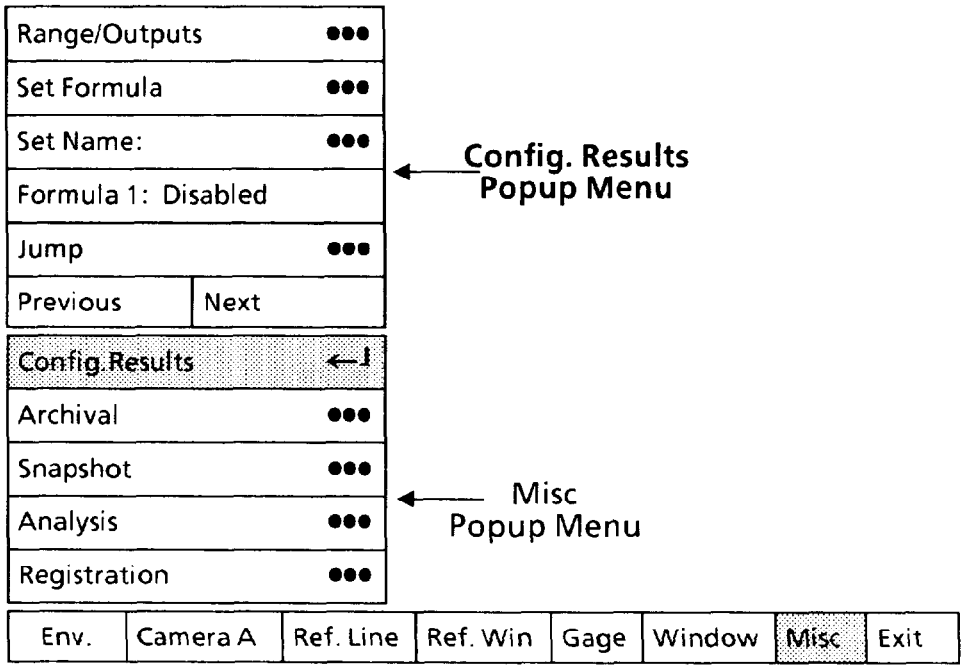
When the first two conditions *are* satisfied, you *can* enable range checking for the selected tool by picking the Range menu box so that it reads "Enabled." Picking the menu box repeatedly causes it to toggle between Enabled and Disabled.

Misc Popup Menu

When you pick the Misc menu box in the Main menu, the Misc popup menu appears with a new menu box called Config. Results.



When you pick the Config. Results menu box in the Misc menu, the Config. Results popup menu box appears. This is the menu that you use to configure MATH-PAK formulas.



Misc Popup Menu (continued)

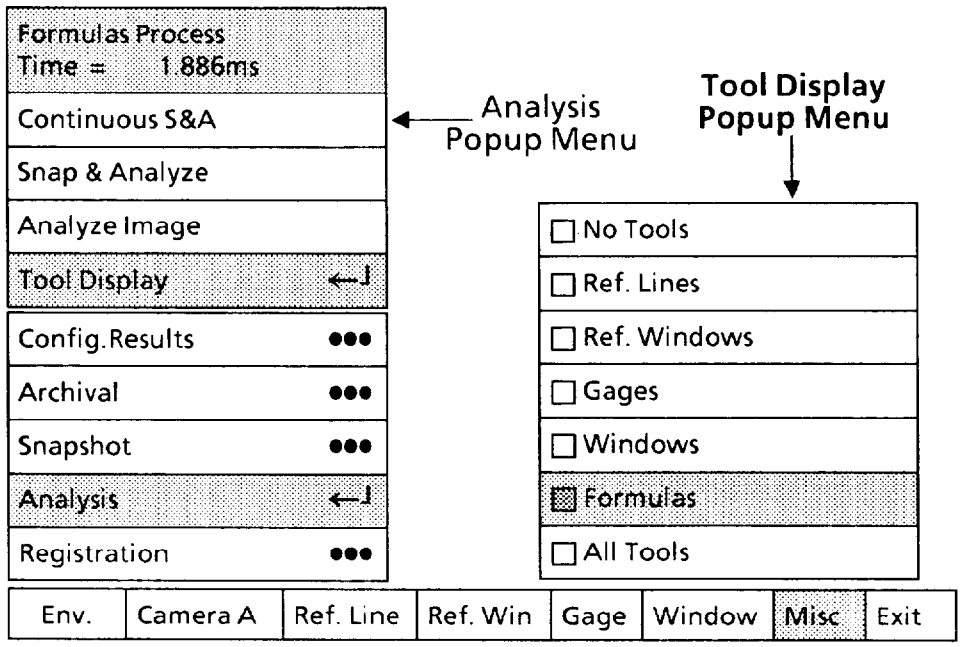
Initially, only the Set Formula, Jump, and Previous/Next menu boxes are accessible (they are in light type). These boxes enable you to select the formula *number* that you want to configure. The other boxes are not accessible at this time (they are in **black type**).

Here is a summary list of the steps you will normally take to configure MATH-PAK formulas:

1. Select the formula number using the Jump, Previous, and/or Next menu boxes.
2. Configure the formula using the Set Formula menu box.
3. Select a formula name using the Set Name menu box.
4. Enable the formula by toggling the Formula # menu box.
5. Set the range limits and select the outputs for the enabled formula using the Range/Outputs menu box.

The *Configuring MATH-PAK Formulas* section of this manual describes these steps in detail.

When you pick the Analysis menu box in the Misc popup menu and the Tool Display menu box in the Analysis popup menu, the Tool Display popup menu appears, as follows:

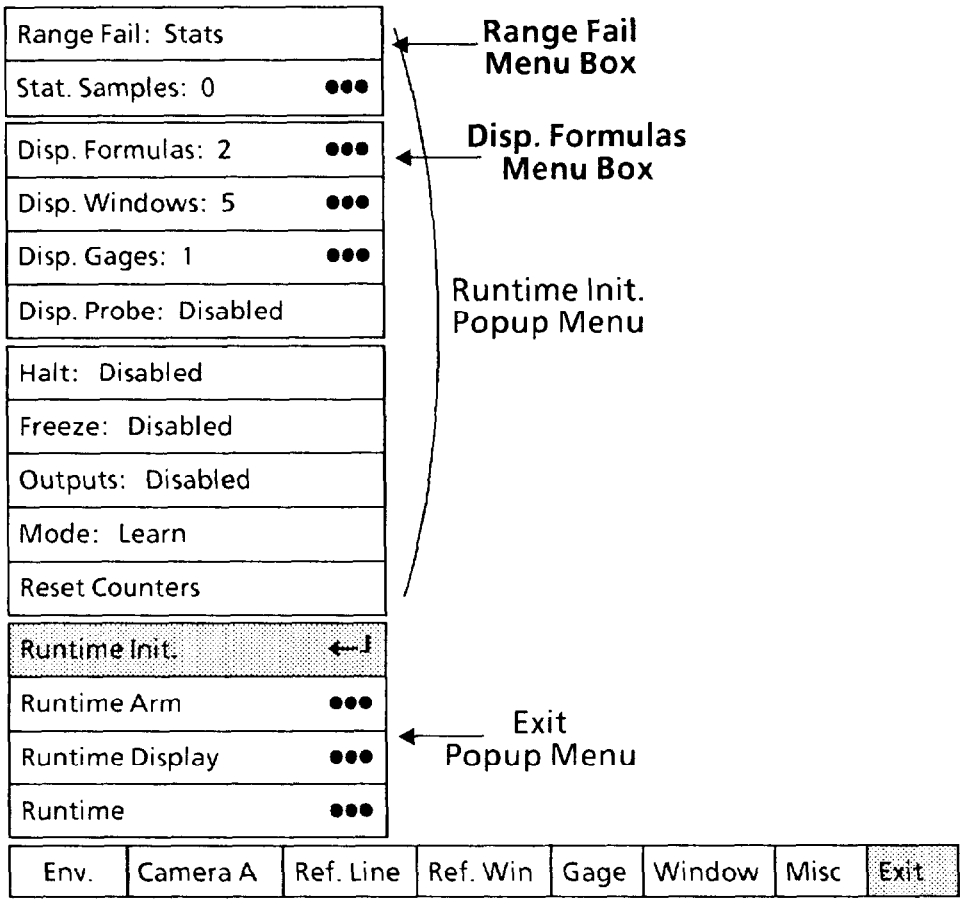


Included in the Tool Display popup menu is the Formulas menu box, which enables you to analyze the process time for all *enabled* formulas.

When you pick Formulas, all *enabled* tools appear on the monitor screen (as they do when you pick All Tools). When you then pick the Continuous S&A, Snap & Analyze, or Analyze Image menu box, the process time for all enabled formulas appears after "Time = " in the box above the Analysis popup menu. (The *color* of the tools indicates their pass/warning/fail status.)

Exit Popup Menu

When you pick the Exit menu box in the Main menu, and then pick the Runtime Init. menu box in the Exit popup menu, the Runtime Init. popup menu appears with two new menu boxes, as follows:



Range Fail Menu Box

The Range Fail menu box gives you two options for calculating inspection results statistics:

1. Using results from *all* inspected parts, including those that *fail* the range checks, *or*

2. Using results *only* from parts that *pass* the range checks.

The Range Fail menu box displays either "Stats" or "No Stats." When "Stats" appears in the menu box, statistics are based both on results that *pass* the range checks and those that *fail*. When "No Stats" appears, statistics are based *only* on results that *pass* the range checks.

If you pick the Range Fail menu box repeatedly, it will toggle between Stats and No Stats.

Disp. Formulas Menu Box

The Disp. Formulas menu box enables you to display formula results and statistics on the Result Page and Stat 1 Page display tables.

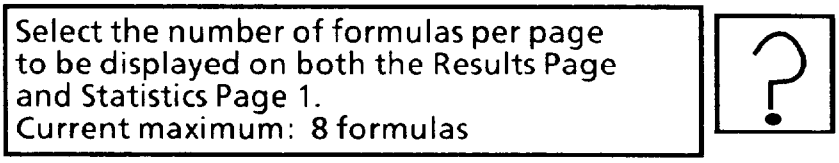
Exit Popup Menu (continued)

Disp. Formulas Menu Box (continued)

By picking the Disp. Formulas menu box, you can specify the number of *formulas* whose results are to be displayed on each page of the Results Page and Stat. Page 1 display tables. (This menu box operates in the same manner as the other "Disp." menu boxes, which are described in Chapter 10 of the *CVIM User Manual*.)

If you pick the *help message icon* . . .

. . . the *help message box* will appear in the upper left corner of the monitor screen and contain a message, such as the following, . . .


Select the number of formulas per page
to be displayed on both the Results Page
and Statistics Page 1.
Current maximum: 8 formulas

. . . that indicates the *current* maximum number of formulas that can appear on each page of the Results Page and Stat 1 Page display tables. The number varies according to how many gages and windows been selected in the Disp. Gages and Disp. Windows menu boxes, and whether or not the light probe results are to be displayed.

Results Page Display

The following illustration is an example showing page *one* of the Results Page display table.

TRIGGERS		FAULTS		Reference Lines / Windows			
				1.	2	1.	1
Accepted:	123456	Master Fault:	4	2.		2.	
Missed:	0	Light Probe:	1	3.		3.	
Total:	123456						

GAGE	Faults	Fail Low	Warn Low	Reading	Warn Hi	Fail Hi
1.	0			127.000		

WINDOW	Faults	Fail Low	Warn Low	Reading	Warn Hi	Fail Hi
1.	0	3000	3100	3182	3300	3400
2.						

FORM.	Faults	Fail Low	Warn Low	Reading	Warn Hi	Fail Hi
1.						
2.						
Y POS 1	0	110.000	120.000	127.000	140.000	150.000
Y POS 2	0	105.000	115.000	124.000	135.000	145.000
X POS 1	0	130.000	140.000	152.000	160.000	170.000

Exit Popup Menu (continued)

Results Page Display (continued)

In the illustration, the data appearing in the GAGE, WINDOW, and FORM. boxes have the following significance:

- Gage #1 is enabled and has returned results data; however, gage #1 results data are not ranged checked because formula #9 (which is on page *two*) is enabled. Thus, fail and warning range limits are not displayed for gage #1.
- Window #1 is enabled and has returned results data. Its results are ranged checked, since formula #1 is not enabled. Window #2 is disabled; thus, no data appear.
- Formulas #1 and #2 are disabled; thus, no data appear. Formula #3 is named Y POS 1. Note that its results are the *same* as the results from gage #1, shown above, but since Y POS 1 is a formula, its results are range checked. Formula #4 is named Y POS 2, and formula #5 is named X POS 1.

Note that the results data from the light probe are *not* displayed in the example (the Disp. Probe menu box reads "Disabled."). If probe results were *enabled* for display, the total number of formulas, gages, and windows displayed would be reduced.

Stat 1 Page Display

The following illustration is an example showing page *one* of the Stat 1 Page display table.

GAGE	Samples	Mean	Std. Dev.	Min.Reading	Max. Reading
1.					
WINDOW	Samples	Mean	Std. Dev.	Min.Reading	Max. Reading
1.	186	3179.105	2.063	3171	3186
2.					
FORM.	Samples	Mean	Std. Dev.	Min.Reading	Max. Reading
1.					
2.					
Y POS 1	186	122.284	0.613	121.000	124.000
Y POS 2	186	133.182	1.741	127.000	137.000
X POS 1	186	152.000	0.000	152.000	152.000

The statistics data appearing in the GAGE, WINDOW, and FORM. boxes have the following significance:

- Gage #1 is enabled; however, gage #1 statistics cannot be accumulated because formula #9 (which is on page *two*) is enabled. Thus, statistics data are not displayed for gage #1.

Exit Popup Menu *(continued)*

Stat 1 Page Display *(continued)*

- Window #1 is enabled and has returned results data. Its results are ranged checked, and, since formula #1 is not enabled, statistics data are displayed for window #1. Window #2 is disabled; thus, no data appear.

- Formulas #1 and #2 are disabled; thus, no data appear. Formula #3 is named Y POS 1, formula #4 is named Y POS 2, and formula #5 is named X POS 1.

Note that statistics data from the light probe are *not* displayed in the example (the Disp. Probe menu box reads "Disabled."). If probe results were *enabled* for display, the total number of formulas, gages, and windows displayed would be reduced.

NOTE: During the run mode, when you select the Results Page (or Stat 1 Page) display table, you must pick the Page ↓ (page down) box one or more times in order to see the results (or statistics) from any formula whose number is *beyond* the number that you selected in the Disp. Formulas menu box.

For example, if you want to see the results from formula #12 and you selected "5" in the Disp. Formulas menu box, formulas #1 through #5 will appear on page *one* of the Results Page display table, formulas #6 through #10 will appear on page *two*, and formulas #11 through #15 will appear on page *three*. Thus, you must pick the Page ↓ box *two times* to reach page three and see the results (or statistics) from formula #12.

I/O Page Display

A *second* page has been added to the I/O Page display table in order to display the I/O status of all 56 formulas. The formula display box on page two provides the same kind of pass/warn/fail status information as the gage and window display boxes on page one. You can access page two by picking the Page ↓ (page down) box once. (The I/O Page display table is described fully in Chapter 10 of the *CVIM User's Manual.*)

All Tools Display

When you select the All Tools display function for the run mode, the names of all of the *enabled* formulas appear on the monitor screen, starting in the upper left corner. On a color monitor, the color indicates the pass/warn/fail condition of each formula.

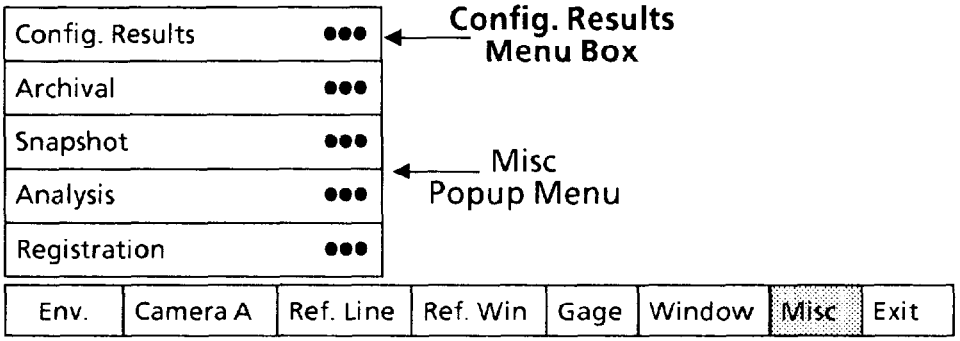
Note that disabled formulas will *not* appear in the screen listing. Thus, if formulas #1 and #56 are the only ones enabled, only those two formulas will appear in the list, with formula #1 on top.

Configuring MATH-PAK Formulas

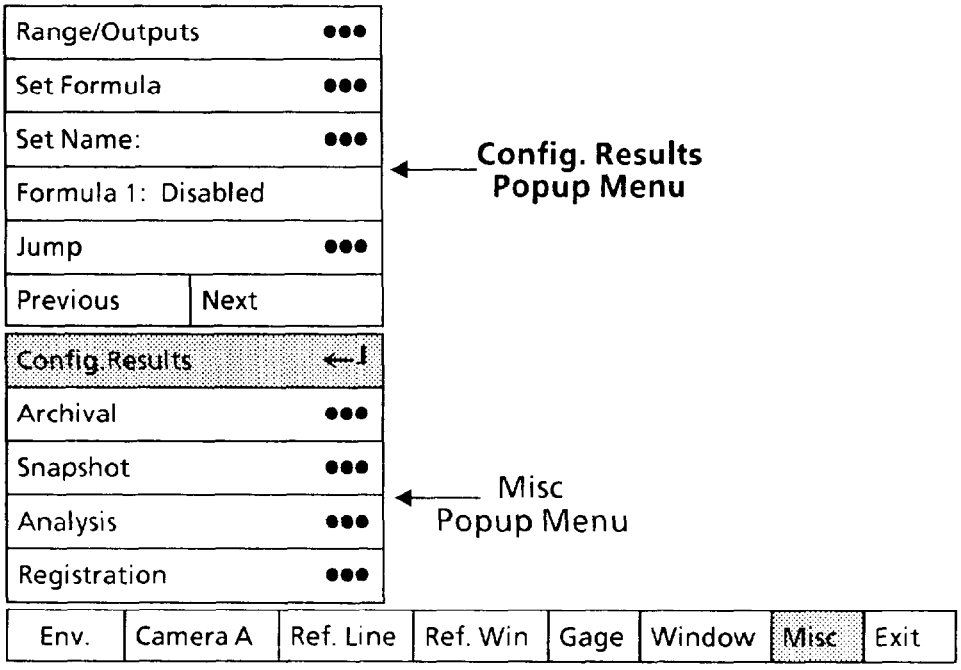
As indicated earlier, the MATH-PAK option adds a menu box called Config. Results, and it is your entry point for configuring MATH-PAK formulas.

Use the following steps to configure MATH-PAK formulas.

Your Action	Comments
Pick the Misc *menu box in the Main Menu.*	When you pick the Misc menu box in the Main Menu, the Misc popup menu shows the Config. Results menu box, as follows:



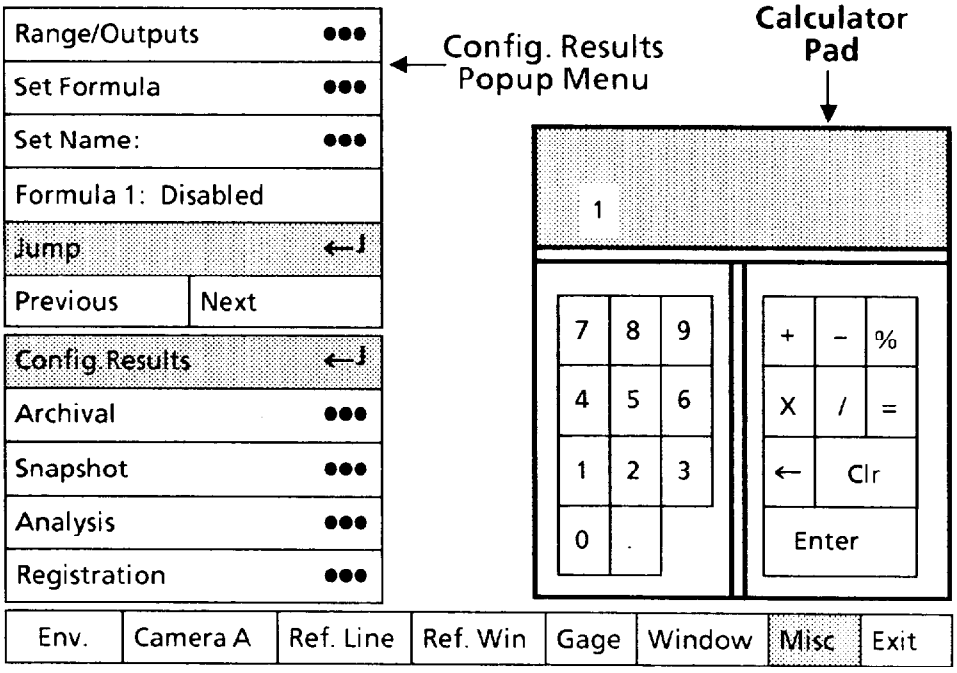
Pick the Config. Results *menu box.*	When you pick the Config. Results menu box, the Config. Results popup menu appears, as follows:



When no formula has been configured for the *current* formula number, the Config. Results popup menu appears as shown above. In this case, the Set Name menu box has no name in it, and the Formula # menu box shows that the selected formula number is Disabled. In addition, the menu boxes labeled Range/Outputs, Set Name, and Formula# appear in **black** type (meaning that you cannot pick them). Only the Set Formula menu box is accessible at this time.

Configuring MATH-PAK Formulas (continued)

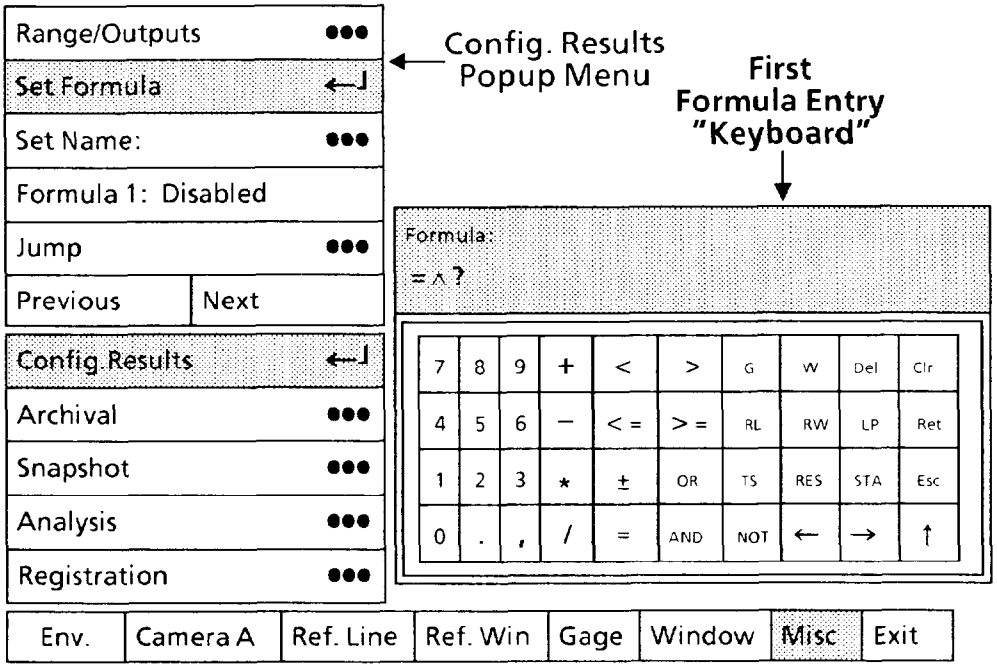
Your Action	Comments
Look at the Formula # *menu box.*	The Formula # menu box shows the selected formula number and its current status (Disabled or Enabled). If the current formula number is *not* the one you want, select the desired formula number by using the Previous or Next function and/or the Jump function, as follows:
Pick the Next *menu box to select the next higher formula number.*	Pick the Next menu box once for each increment in the formula number.
	or,
Pick the Previous *menu box to select the next lower formula number.*	Pick the Previous menu box once for each decrement in the formula number.
	or,
Pick the Jump *menu box.*	When you pick the Jump menu box, a "calculator pad" appears on the monitor screen, as follows:



Pick each digit of the new formula number.	As you pick each digit, it appears in the calculator "display."
Pick the Enter *key.*	When you pick the Enter key, the new formula number in the calculator display appears in the Formula # menu box. This means that the new formula number has been selected.

Configuring MATH-PAK Formulas (continued)

Your Action	Comments
Pick the Set Formula *menu box.*	When you pick the Set Formula menu box, the *first* of two formula entry "keyboards" appears on the monitor screen:

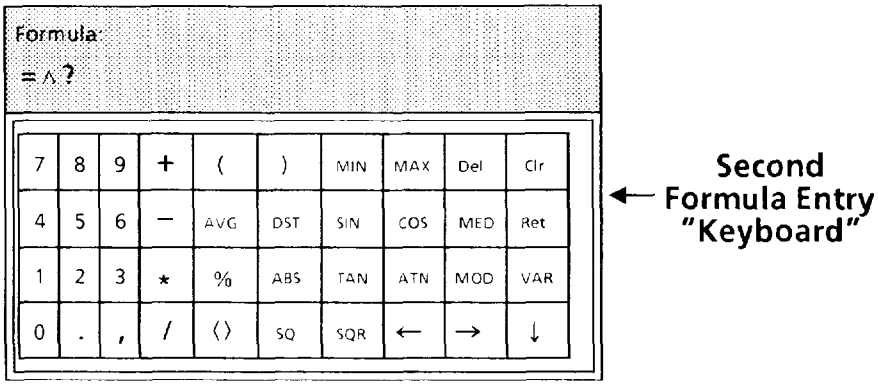


The upper box in the keyboard displays the name of the currently selected formula and the formula itself. In this case, no formula has been configured nor formula name entered.

The lower box in the keyboard displays the "keys" that are used to select the digits (0 - 9), the mathematical, logical, and tool operators, and the operands. The remaining keys are for editing and saving formulas.

Pick the up arrow (↑) in the lower-right corner.

When you pick the up arrow (↑), the up arrow changes to a down arrow (↓), and a *second* formula entry keyboard appears in which the rightmost six columns of keys have different functions than in the first keyboard:



Configuring MATH-PAK Formulas (continued)

As stated earlier, most of the keys in both keyboards are used to select the various elements that make up a formula. The remaining keys are used to edit and store a formula.

The next four sections contain the following information:

- *Definitions: MATH-PAK Terms and Concepts* describes the key terms and concepts applicable to MATH-PAK formulas.
- *Definitions: Formula Entry Key Functions* describes briefly the specific functions of the keys in each category: control, operand, and operator keys.
- *Using Control Keys* describes, in detail, the procedure for using each control key. Simple examples are included to illustrate the procedures.
- *Using Operator and Operand Keys* describes, in detail, the procedure for using each operand and operator key. Simple examples are included, where appropriate, to illustrate these procedures.

Definitions: MATH-PAK Terms and Concepts

This section briefly defines some of the key terms and concepts as they apply to the MATH-PAK option.

MATH-PAK formulas consist of combinations of operators and operands that act together to produce a result.

- *Operator:* a symbol (such as "+") or a term (such as "AND") that represents a mathematical or logical operation upon one or more operands. (All of the operators are listed under the *Operator Keys* heading.)

  Operators are further classified as *arithmetic*, *logical*, and *conditional*, as follows:

  Arithmetic operators, such as +, –, *, ÷, and TAN perform addition, subtraction, multiplication, division, tangent, and other operations.

  Logical operators, such as AND, along with *conditional* operators, such as <>(not equal), produce Boolean results, where the results are always either *true* (logic 1) or *false* (logic 0).

- *Operand:* a quantity or value (such as a constant, a tool result, or a result from a formula operation) upon which a mathematical or logical operation is performed. (Operands are listed in a table under the *Operand Keys* heading.)
- *Result:* the value generated by operators acting upon operands.

The following examples demonstrate the use of various operands and operators, and the results they produce, in arithmetic and logical formulas.

Definitions: MATH-PAK Terms and Concepts *(continued)*

Arithmetic examples: Here are a few examples of how the terms *operand*, *operator* and *result* are used within various arithmetic formula constructions:

(1) GAGE1 + GAGE2

In this formula, GAGE1 and GAGE2 are *operands* that represent results data from gages #1 and #2, and + is an arithmetic *operator*.

(2) (3 * WIN1) + WIN2

In this formula, (3 * WIN1) and WIN2 are both *operands*, and + is an arithmetic *operator*.

The expression (3 * WIN1) is *itself* an *operand* representing the multiplication of the window #1 results data by the constant 3. WIN1 and 3 are operands in the expression, and * is an arithmetic *operator*.

(3) MAX(GAGE1,WIN1,750)

In this formula, GAGE1,WIN1 and 750 are all *operands*, and MAX is an arithmetic *operator*.

In the expression, the MAX operator causes the values of the three operands (gage #1 and window #1 measurements, and the constant 750) to be compared. The formula result will be the *largest* value.

Logic examples: Results from logic formulas are either *true* (logic 1), if the logic condition is satisfied, or *false* (logic 0), if the logic condition is *not* satisfied.

NOTE: From the standpoint of the *discrete outputs*, a "true" (logic 1) result is equivalent to a "pass" condition, and a "false" (logic 0) is equivalent to a "fail" condition.

Here are a few examples of how the terms *operand*, *operator* and *result* are used within various logical formula constructions:

(1) GAGE1 < 5

In this formula, the left operand (the measurement result from gage #1) is compared to the right operand (the constant 5). If the gage #1 result is less than 5, the formula result will be *true* (logic 1). If the gage #1 result is equal to or greater than 5, the formula result will be *false* (logic 0).

(2) GAGE1 > GAGE2

In this formula, the left operand (the measurement result from gage #1) is compared to the right operand (the measurement result from gage #2). If the gage #1 result is greater than the gage #2 result, the formula result will be *true* (logic 1). If the gage #1 result is equal to or less than the gage #2 result, the formula result will be *false* (logic 0).

Definitions: MATH-PAK Terms and Concepts *(continued)*

(3) GAGE1 = (GAGE2/4)

In this formula, the left operand (the measurement result from gage #1) is compared to the right operand (the expression in which the gage #2 measurement result is divided by 4). If the gage #1 result equals the *expression* result, the formula result will be *true* (logic 1). If the gage #1 result is *not* equal to the expression result, the formula result will be *false* (logic 0).

(4) GAGE1.1 AND GAGE2.1

In this formula, the left operand (the warning range result from gage #1) is ANDed with the right operand (the warning range result from gage #2). These operands will *always* be either *true* (logic 1) or *false* (logic 0).

If both warning range results are logic 1, the formula result will be *true* (logic 1). If either warning range result is logic 0, the formula result will be *false* (logic 0).

(5) (GAGE1 = 7) OR (GAGE1 = 12)

In this formula, the left operand (the *first* expression) compares the measurement result from gage #1 to 7. The expression result is then ORed with the right operand (the *second* expression), which compares the gage #1 result with 12. The expression result will *always* be either *true* (logic 1) or *false* (logic 0).

If either measurement result is logic 1, the formula result will be *true* (logic 1). If both measurement results are logic 0, the formula result will be *false* (logic 0).

Definitions: Formula Entry Key Functions

This section briefly defines the function of each formula entry key. The *Using Control Keys* and *Using Operator and Operand Keys* sections describe the use of each key in detail.

Digit Keys: 0 to 9

These keys are used to enter all of the numeric values to be used in a formula.

Here are some limitations on entering and using *constants*:

- The largest *positive integer* that the system can accept is 32,767. The system blocks any attempt to enter a larger positive integer.
- The largest *negative integer* that the system can accept is -32,767. The system blocks any attempt to enter a "larger" negative integer.
- The largest *positive fixed point* number that the system can accept is 32,767.999. The system blocks any attempt to enter a larger positive fixed point number.
- The largest *negative fixed point* number that the system can accept is -32,767.999. The system blocks any attempt to enter a "larger" negative fixed point number.

Definitions: Formula Entry Key Functions (continued)

Operand Keys

These keys are used to enter operands (variable data) into a formula. When entered into a formula, these operands can supply inspection results data from specific analysis tools, results data from the operation of specific formulas, and statistical data from the operation of specific formulas.

The following table lists these operands and briefly describes their functions.

Key	Operand Name	Operand Function
TS	Tool set change	Used with the following operands (G, W, RL, etc.) to obtain results data from the alternate (non-active) tool set.
G	Gage	Provides inspection results data from one of the 32 gages.
W	Window	Provides inspection results data from one of the 24 windows.
RL	Reference line	Provides inspection results data from one of the three reference lines.
RW	Reference window	Provides inspection results data from one of the three reference windows.
LP	Light probe	Provides inspection results data from the light probe.
RES	Results	Provides results data from the operation of one of the 56 possible formulas or inspection tools.
STA	Statistics	Provides statistical data from the operation of one of the 56 possible tools and formulas.
0 - 9	Digits	Provide integer or fixed point constants in a formula.

The operands are further defined according to the specific *type* of results data to be supplied to a formula. As an example, you can specify that a particular gage operand (one of 32) supply its basic measurement value (number of pixels, number of edges, and so on), *or* its pass/fail result for the warning *or* fault range (1 or 0), *or* a second coordinate (for X and Y position functions only).

You can specify results data in a formula by appending a *decimal point* and a *number* to the operand number(s). For example, if you select the gage #1 operand, GAGE1.1 returns the pass/fail result for the warning range, GAGE1.2 returns the pass/fail result for the fault range, and so on.

NOTE: Some inspection results are always returned in *fixed point* format (linear gaging, luminance), while others are always returned in *integer* format (# white pixels, # edges). In all cases, if *one* of the operands in a formula returns fixed point results, the formula will always return a fixed point result.

Definitions: Formula Entry Key Functions (continued)

Operand Keys (continued)

The following table lists the key code and subcodes for each operand, along with a brief description of the type of results data that each code/subcode combination returns when used within a formula.

Key Code	*Operand Name	*Sub-code (s)	Type of Results Data Returned to Formula
G	GAGEn.s (n = 1 to 32) (s = 0 to 3)	n.0	Gage measurement value (# of pixels, # of edges, etc.).
		n.1	Pass/fail condition for warning range (1 = pass, 0 = fail).
		n.2	Pass/fail condition for fault range (1 = pass, 0 = fail).
		n.3	Second coordinate value (Y coordinate for X Position gage; X coordinate for Y Position gage).
W	WINn.s (n = 1 to 24) (s = 0 to 2)	n.0	Window measurement value (# of pixels, # of objects, etc.).
		n.1	Pass/fail condition for warning range (1 = pass, 0 = fail).
		n.2	Pass/fail condition for fault range (1 = pass, 0 = fail).
RL	REFLn.s (n = 1 to 3) (s = 0 to 3)	n.0	X coordinate value of edge position on reference line.
		n.1	Y coordinate value of edge position on reference line.
		n.2	Pass/fail condition for reference line (1 = pass, 0 = fail).
		n.3	Theta from a reference line configured for "X-X then Y" or "Y-Y then X" rotation compensation.
RW	REFWn.s (n = 1 to 3) (s = 0 to 15)	n.0	Cumulative position: X coordinate value.
		n.1	Cumulative position: Y coordinate value.
		n.2	Cumulative angle (θ).
		n.3	Cumulative pass/fail condition (1 = pass, 0 = fail).
		n.4	Active feature #1 position: X coordinate value.
		n.5	Active feature #1 position: Y coordinate value.
		n.6	Active feature #1 score value.
		n.7	Active feature #1 pass/fail condition (1 = pass, 0 = fail).
		n.8	Active feature #2 position: X coordinate value.
		n.9	Active feature #2 position: Y coordinate value.
		n.10	Active feature #2 score value.
		n.11	Active feature #2 pass/fail condition (1 = pass, 0 = fail).
		n.12	Active feature #3 position: X coordinate value.
		n.13	Active feature #3 position: Y coordinate value.
		n.14	Active feature #3 score value.
		n.15	Active feature #3 pass/fail condition (1 = pass, 0 = fail).
LP	LPRB.s (s = 0 to 2)	.0	Light probe brightness value.
		.1	Pass/fail condition for warning range (1 = pass, 0 = fail).
		.2	Pass/fail condition for fault range (1 = pass, 0 = fail).
RES	RSLTn.s (n = 1 to 56) (s = 0 to 2)	n.0	Formula (n) results data value.
		n.1	Pass/fail condition for warning range (1 = pass, 0 = fail).
		n.2	Pass/fail condition for fault range (1 = pass, 0 = fail).
STA	STATn.s (n = 1 to 56) (s = 0 to 4)	n.0	Formula/tool (n) statistics: Number of samples taken.
		n.1	Formula/tool (n) statistics: Minimum reading value.
		n.2	Formula/tool (n) statistics: Maximum reading value.
		n.3	Formula/tool (n) statistics: Mean value.
		n.4	Formula/tool (n) statistics: Standard deviation value.

*The letter n = operand number; s = operand subcode. The Operand Name column shows how each operand appears when entered into a formula.

Definitions: Formula Entry Key Functions (continued)

Operator Keys

The operator keys are used to select the mathematical, logical, and other operators in a formula. The next two tables list these operators and briefly describe their functions.

Key	Name	*Result Type	Operator Functions (Part 1 of 2)
ABS	Absolute value	INT/FP	Computes the absolute value of a number (negates the value if it is negative).
ATN	Arc tangent	FP	Computes an angle (in degrees) from operands Δ X and Δ Y, which are the *changes* in the X and Y coordinates of a point (such as an edge) that has moved within the image field.
AVG	Average	INT/FP	Computes the average of a list of operands.
COS	Cosine	FP	Computes the cosine of an angle (in degrees). Returns a fixed point number.
DST	Distance	FP	Computes distance, given the four operands X_1, Y_1, X_2, and Y_2, by computing the square root of the sum of the squared differences.
MAX	Maximum	INT/FP	Finds the *maximum* numeric value in a list of operands.
MED	Median	INT/FP	Finds the middle numeric value in a list of operands by sorting the list in ascending order and returning the value located at the center position in the sorted list.
MIN	Minimum	INT/FP	Finds the *minimum* numeric value in a list of operands.
MOD	Mode	INT/FP	Finds the value in a list that appears most often. If no value appears more than once, MOD returns the lowest value.
SIN	Sine	FP	Computes the sine of an angle (in degrees). Returns a fixed point number.
SQR	Square root	FP	Computes the square root of the operand.
SQ	Square	INT/FP	Computes the square of the operand.
TAN	Tangent	FP	Computes the tangent of an angle (in degrees). Returns a fixed point number.
VAR	Variance	FP	Computes the variance of a list of operands. The variance is equal to the sum of the square of the difference between each operand in the list and the average value of the operands, divided by the number of operands in the list.
(	Open parenthesis	INT/FP	Used to set the order of evaluation of a formula. Can be nested.
)	Closed parenthesis	INT/FP	Used to close an open parenthesis.
+	Add	INT/FP	Adds the right operand to the left operand.
–	Subtract	INT/FP	Subtracts the right operand from the left operand.
*	Multiply	INT/FP	Multiplies the left operand by the right operand.
/	Divide	INT/FP	Divides the left operand by the right operand.
%	Modulo	INT/FP	Produces the remainder of the division of the left operand by the right operand.

*Result types: INT = integer; FP = fixed point; INT/FP = either integer or fixed point.

Definitions: Formula Entry Key Functions (continued)

Operator Keys (continued)

Key	Name	*Result Type	Operator Functions (Part 2 od 2)
±	Unary minus	INT/FP	Negates a numeric value.
.	Decimal point	FP	Used to enter a fixed point number. Also used to designate an operand subtype.
=	Equal	INT/FP	Produces a "1" result if the left operand is *equal to* the right operand; otherwise, produces a "0" result.
⟨⟩	Not equal	INT/FP	Produces a "1" result if the left operand is *not equal to* the right operand; otherwise, produces a "0" result.
>	Greater than	INT/FP	Produces a "1" result if the left operand is *greater than* the right operand; otherwise, produces a "0" result.
<	Less than	INT/FP	Produces a "1" result if the left operand is *less than* the right operand; otherwise, produces a "0" result.
> =	Greater than or equal	INT/FP	Produces a "1" result if the left operand is *greater than or equal to* the right operand; otherwise, produces a "0" result.
< =	Less than or equal	INT/FP	Produces a "1" result if the left operand is *less than or equal to* the right operand; otherwise, produces a "0" result.
,	Comma	INT/FP	Used to separate the operands in a list of operands.
AND	Logical AND	INT/FP	Produces a "1" result if *both* the left and right operands are *nonzero*; otherwise, produces a "0" result.
OR	Logical OR	INT/FP	Produces a "1" result if *either* the left or right operand is *nonzero*; otherwise, produces a "0" result.
NOT	Logical NOT	INT/FP	Used to *invert* a logical result.

***Result types: INT = integer; FP = fixed point; INT/FP = either integer or fixed point.**

Control Keys

Both keyboards contain the same set of control keys, whose functions are briefly described in the following table.

Key	Name	Control Key Functions
Del	Delete item	Deletes the formula item to the *left* of the cursor.
Clr	Clear formula	Clears the entire formula.
Ret	Return	Causes the formula to be saved in memory.
Esc	Escape	Restores the formula to its condition when last saved.
←	Left arrow	Moves the cursor one item to the left.
→	Right arrow	Moves the cursor one item to the right.
↓	Down arrow	Selects the *second* keyboard from the *first* keyboard.
↑	Up arrow	Selects the *first* keyboard from the *second* keyboard.

Using Control Keys

The control keys enable you to edit a formula and store it in memory.

This section describes the use of the control keys and provides the requirements and limitations for using them. The keys are discussed in the order of their appearance in the preceding table.Where appropriate, an example is provided to illustrate the use of a key.

It is assumed at this point that you have selected an unused formula number, and the first formula entry keyboard is on the screen, as described earlier in the *Configuring MATH-PAK Formulas* section.

Del (Delete) Key

The Del key operates as a backspace/delete key. It can delete an item previously entered into a formula. When you pick the Del key, the CVIM system deletes the item to the left of the cursor (∧) and moves the cursor one space to the left.

To use the Del key, you must first position the cursor (using the left arrow (←) or right arrow (→) key) to the immediate *right* of the formula item to be deleted. You then pick the Del key to delete the item from the formula.

The following procedure shows you how to use the Del key. The example shows the entry of a formula that adds two integers, then changes the formula to subtract two integers by changing the add (+) operator to a subtract (–) operator.

Your Action	Comments
Select either formula entry keyboard.	The Del key is on both keyboards.
Pick these three keys: 4 + 2.	This enters a simple "formula" for adding 4 and 2. Note that the cursor (∧) is to the right of the number 2.

```
Formula:
= 4 + 2∧
```

Now, suppose you want the formula to subtract 2 from 4. This means you must change the add (+) operator to a subtract (–) operator. Continue as follows.

Pick the left arrow (←) key. This moves the cursor (∧) to the left one space. It should now be to the left of the add (+) operator:

```
Formula:
= 4 +∧ 2
```

Your Action	Comments
Pick the Del *(delete) key.*	This *deletes* the add (+) operator and moves the cursor (∧) to the right of the number 4:

```
Formula:
= 4∧ ? 2
```

NOTE: Although the Del key can delete formula items from the box, it does *not* delete the item from memory *if the formula was previously saved using the return* (Ret) *key.* If you change your mind after picking the Del key, you can restore the entire last-saved formula by picking the Esc key.

Pick the – (minus) key. This enters the subtract (–) operator into the formula:

```
Formula:
= 4 –∧ 2
```

The formula is now reconfigured to subtract 2 from 4.

Clr (Clear) Key

The Clr (clear) key can delete an entire formula. The cursor (∧) position is not important.

To use the Clr key, just pick the key. When you do, all items currently entered in the formula will disappear, and the box will be restored to its "clear" state:

```
Formula:
=∧ ?
```

NOTE: Although the Clr key deletes the entire formula from the box, it does *not* delete it from memory *if the formula was previously saved using the return* (Ret) *key.* If you change your mind after picking the Clr key, you can restore the entire last-saved formula by picking the Esc key.

Using Control Keys (continued)

Ret (Return) Key

The Ret (return) key performs two functions at the same time: It performs a computation using a *correctly entered* formula, and it stores the formula in the CVIM system's memory.

NOTE: When you pick the Ret key, a *previously entered* formula is *overwritten.*

To use the Ret key, be sure the formula is entered correctly, to the best of your knowledge, then pick the Ret key. Using the example under the Del key, the result of the computation appears in the box as follows:

```
Formula:                                    result = 2
= 4 -^ 2
```

At the same time, a message box appearing in the upper left corner of the screen indicates that the formula has been saved in memory:

```
Formula saved.
```

At this point, a *previously entered* formula will have been overwritten by the new formula.

If you pick the Ret key with an *incomplete* or *incorrectly entered* formula, the CVIM system will respond with this message:

```
WARNING: A formula error will cause deletion.
Reselect to confirm.
```

NOTE: Before picking the Ret key, check the formula for a question mark (?). If the formula is *correct* (so far as the system is concerned), the question mark will *not* be present.

Using Control Keys (continued)

Esc (Escape) Key

The Esc (escape) can restore the last-saved formula. It deletes all items entered since the last time you picked the Ret key.

Thus, if your formula appeared like this the last time you picked the Ret key . . .

```
Formula:                                    result = 2
= 4 - 2^
```

. . . and you enter the following items (but do *not* pick the Ret key after entering them) . . .

```
Formula:
= 4 - 2 + 6 + 12^
```

. . . and then pick the Esc key, the formula will delete the new items and return to its previous state:

```
Formula:                                    result = 2
= 4 - 2^
```

Cursor Keys: (←) and (→)

These keys can reposition the cursor (^) within a formula without deleting any item in the formula. For example, when you pick the left arrow key (←), the cursor moves one item to the left. If your formula looks like this initially . . .

```
Formula:
= 4 - 2 + 6 + 12^
```

. . . and you pick the left arrow key, the cursor will move to the left of the add (+) operator:

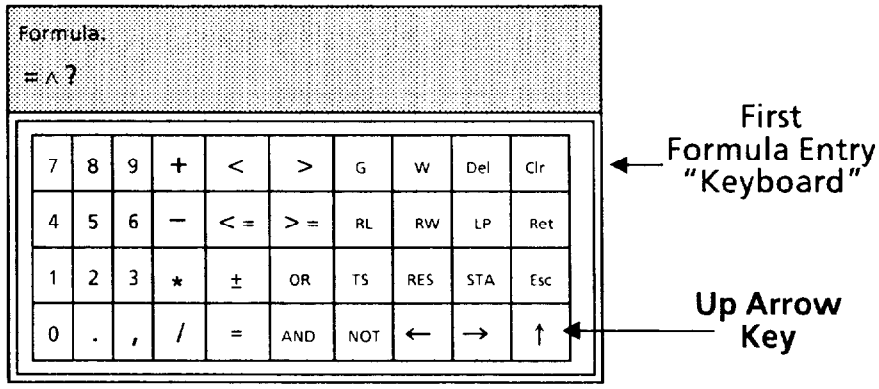
```
Formula:
= 4 - 2 + 6 +^ 12
```

Using Control Keys (continued)

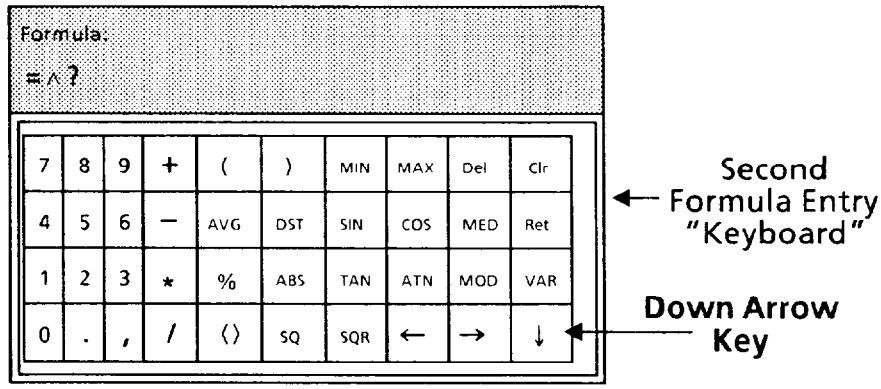
Keyboard Select Keys: (↑) and (↓)

The up arrow (↑) and down arrow (↓) are used to display the *alternate* formula entry keyboard.

The up arrow (↑) appears on the *first* keyboard – this is the one that appears right after you pick the Set Formula menu box:



When you pick the up arrow key, the *second* keyboard appears. This keyboard contains the down arrow (↓):


Formula:
=^?
7 8 9 + ( ) MIN MAX Del Clr
4 5 6 − AVG DST SIN COS MED Ret
1 2 3 * % ABS TAN ATN MOD VAR
0 . , / () SQ SQR ← → ↓
Second Formula Entry "Keyboard"
Down Arrow Key

You may need to alternate between the two keyboards occasionally as you enter your MATH-PAK formulas.

Using Operator and Operand Keys

This section provides detailed information for using the operator and operand keys to enter formulas. Most of these keys require that you observe certain procedures and limitations in order to use them correctly.

Note that the limitations applying to the digit keys (0 to 9), when these keys are used to enter constant numeric values, were described earlier under the *Digit Keys: 0 to 9* heading.

The following operator key procedures are presented in the order of their appearance in the preceding *Operator Function* tables. These procedures use example formulas to illustrate using the operator and operand keys, and to suggest possible applications.

For the purpose of these procedures, it is assumed that you have just selected an *unconfigured* formula number and picked the Set Formula menu box, which displays a *clear* formula entry box and the *first* formula entry keyboard on the monitor screen (as described earlier under the *Keyboard Select Keys* heading).

NOTE: In the following procedures, you will occasionally need to *alternate* between the two keyboards in order to configure some of the formulas. You can select the alternate keyboard by picking the up arrow (↑) or down arrow (↓) in the lower-right corner of the keyboard.

ABS Key

The ABS (absolute value) function converts a negative number to its absolute value. Thus, –45 is converted to 45.

In a MATH-PAK formula, the ABS operator requires one operand. The basic formula takes this form:

ABS(value)

The following procedure shows you how to set up a formula using the ABS operator. The procedure uses a negative integer to demonstrate the entry of an ABS formula.

Your Action	Comments
Select the second formula entry keyboard.	The ABS operator is on this keyboard.
Pick the ABS *key.*	When you pick the ABS key, the following entry appears in the upper box of the keyboard:

```
Formula:
= ABS(^ ?
```

Note that the open parenthesis symbol, "(", appears with the ABS operator.

Using Operator and Operand Keys (continued) — **ABS Key (continued)**

Your Action	Comments
Select the first formula entry keyboard.	The "unary" operator (±) is on this keyboard.
Pick the (±) key.	When you pick the ± key, the following entry appears in the display:

```
Formula:
= ABS( -^ ?
```

This will *negate* the value that follows it.

Pick these two keys: 4 5.	This enters the value (–45) in the formula. The display should now appear as follows:

```
Formula:
= ABS ( - 45 ^ ?
```

Select the second formula entry keyboard.	The parenthesis keys are on this keyboard.
Pick the "close parenthesis" key, ")".	This completes the ABS formula entry.
Pick the Ret *(return) key.*	This causes the system to display the "result." Note that the original value, –45, is converted to the *absolute* value 45.

```
Formula:                                  result = 45
= ABS ( - 45 )^
```

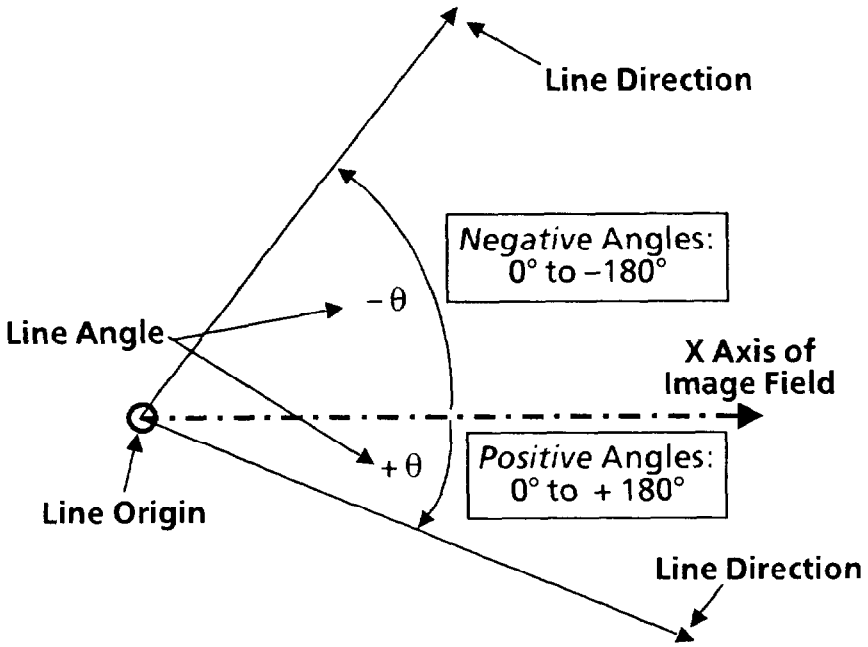
Look at the message box in the upper left corner of the screen.	The "Formula saved" message indicates that the CVIM system has saved the ABS formula in its memory.

This completes the procedure for entering and saving the ABS formula.

Using Operator and Operand Keys *(continued)*

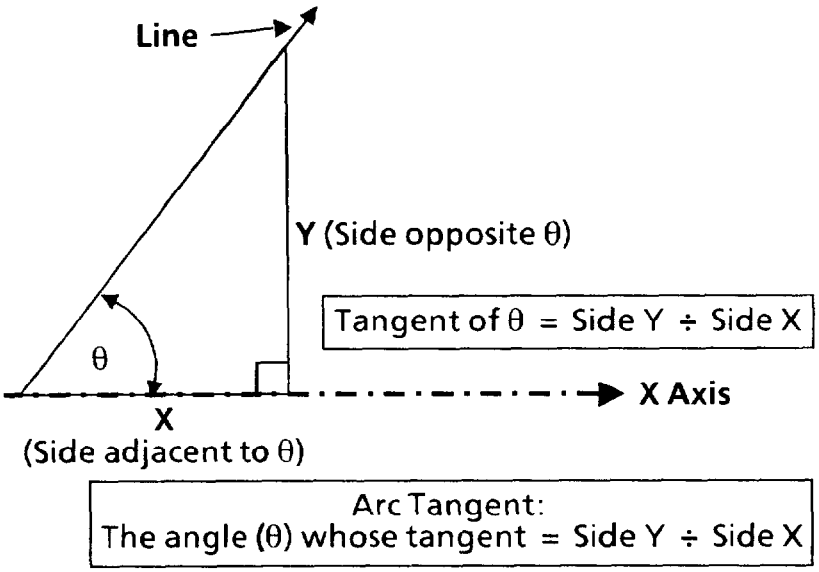
ATN Key

The ATN operator computes the angle θ (in degrees) of a line *relative to the X axis of the image field*, as follows:



The ATN function computes the angle θ by first computing the *tangent* of the angle.

The *tangent* of unknown angle θ (in a right angle triangle) is the *ratio* between the side *opposite* the angle and the side *adjacent* to the angle . . .



. . . and the *arc tangent* is the angle (θ) whose tangent is equal to a given number. In the illustration above, that number is the length of Y divided by the length of X.

Using Operator and Operand Keys *(continued)*

ATN Key *(continued)*

In order to compute the tangent of θ, and from that the *arc tangent* (θ itself), the ATN operator must be given the length of the two sides, **Y** and **X**, of the right triangle formed between a line and the X axis of the image field.

Thus, if you can supply the length of **Y** and **X**, the ATN function can compute the *tangent* of the unknown angle θ and, consequently, the *arc tangent* – the angle θ itself.

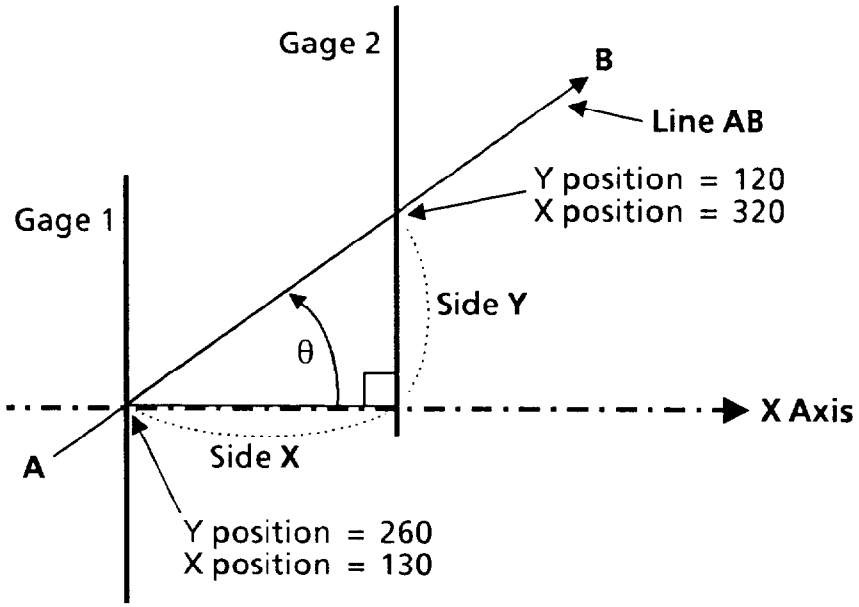
In a MATH-PAK formula, the ATN operator requires the two operands, ΔY and ΔX. The basic formula takes this form:

ATN(ΔY,ΔX)

The ΔY operand represents the length of side **Y**, and the ΔX operand represents the length of side **X**. Values for ΔY and ΔX must be computed and made available to the ATN formula.

Example: The objective of this example is to use the ATN operator to determine the angle θ between line AB and the X axis in the illustration below.

The first step is to find ΔY and ΔX. For this purpose, two linear gages are configured for the "Y Position" gaging operation and positioned *vertically* across line AB, as follows:



Note that the Y position returned by Gage 1 is 260 (pixels), and the Y position returned by Gage 2 is 120. The *difference* (or delta, Δ) is –140, since Y has "moved" from 260 to 140.

Note also that each gage can return the *second* position (in this case, the *X* position). The X position returned by Gage 1 is 130 (pixels), and the X position returned by Gage 2 is 320. The *difference* (or delta, Δ) is 190, since X has "moved" from 130 to 320.

Using Operator and Operand Keys *(continued)*

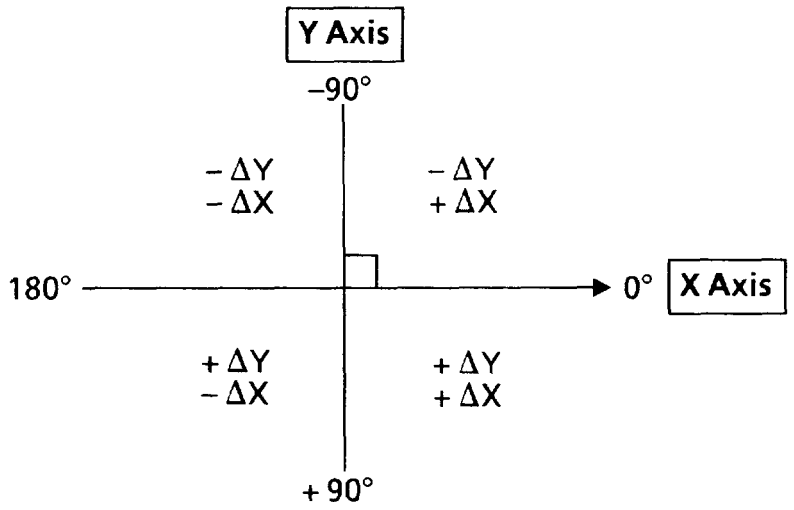
ATN Key *(continued)*

By altering the basic formula to *compute* ΔY and ΔX from the *differences* in the two X and two Y positions, the formula looks like this:

ATN(GAGE2 - GAGE 1, GAGE 2.3 - GAGE 1.3)

In this formula, GAGE2 - GAGE 1 returns ΔY, and GAGE 2.3 - GAGE 1.3 returns ΔX. The ΔY value is –140, and the ΔX value is +190.

In general, the *signs* of ΔY and ΔX relate to the *quadrant* in which the angle lies as follows:



Since ΔY is *negative* and ΔX is *positive* in this example, the ATN function calculates θ to be a negative angle, –36.3°, which in the 1st quadrant.

NOTE: *The ATN operator in this example uses the same frame of reference for measuring an angle that the reference windows and reference lines use for measuring rotation angles.*

The following procedure shows you how to set up a formula using the ATN operator. The procedure uses the preceding formula, ATN(GAGE2 - GAGE 1, GAGE 2.3 - GAGE 1.3), to demonstrate the entry of an ATN formula.

Your Action	Comments
Select the second formula entry keyboard.	The ATN operator is on this keyboard.
Pick the ATN *key.*	When you pick the ATN key, the following entry appears in the upper box of the keyboard:

```
Formula:
= ATN(^ ?
```

Using Operator and Operand Keys *(continued)*

ATN Key *(continued)*

Your Action	Comments
Select the first formula entry keyboard, then pick the "G" (for Gage) key.	This is the first step in assigning a gage operand. The display should appears as follows:

```
Formula:
= ATN ( ? GAGE0^
```

Pick the "2" key.	This assigns gage #2 as the first operand. When configured as GAGE2, this operand returns gage #2 *measurement data* to the formula. In this example, it returns Y position data, since gage #2 would have been configured earlier for that gage operation.

```
Formula:
= ATN ( GAGE2^ ?
```

Pick these four keys: -(minus) G 1 ,(comma).	This sets up gage #1 as the second operand, and thereby completes the requirement for "ΔY" in the ATN formula. Gage #1 returns Y position data.

```
Formula:
= ATN ( GAGE2 - GAGE1, ^ ?
```

Pick these four keys: G 2 .(decimal) 3.	This sets up gage #3 as the third operand. The "2.3" causes gage #2 to return *second coordinate* data to the formula. In this case, it is X position data. (For more information, refer to the "result data types" table under the *Operand Keys* heading.)

```
Formula:
= ATN ( GAGE2 - GAGE1,  GAGE2.3^ ?
```

Using Operator and Operand Keys (continued)

ATN Key (continued)

Your Action	Comments
Pick these five keys: -*(minus)* G 1 .*(decimal)* 3.	This sets up gage #1 as the fourth operand, and thereby completes the requirement for the "ΔX" in the ATN formula. In this case, the "1.3" causes gage #1 to return *second coordinate* data to the formula. Since gage #1 is configured for the Y position gage operation, it returns *X position* data to the formula.

```
Formula:
= ←GAGE2 - GAGE1, GAGE2.3 - GAGE1.3^ ?
```

NOTE: A left arrow (←) has appeared at the left side of the display, and the "ATN (" has disappeared. This indicates that the remainder of the formula has been *shifted off* the display because of space limitations.

Generally, any time a formula is too large to be displayed completely, a left or right arrow appears in the left or right side of the display, indicating the location of the remainder of the formula.

Select the second formula entry keyboard.	The parenthesis keys are on this keyboard.
Pick the "close parenthesis" key, ")".	This completes the ATN formula entry.

```
Formula:
= ←GAGE2 - GAGE1, GAGE2.3 - GAGE1.3 )^
```

Pick the Ret *(return) key.*	This causes the system to display a "result"; however, the number displayed is *not correct* since the "GAGEn.3" operands cannot operate correctly *except in the run mode.* Thus, you should disregard the "result" number in this case.

```
Formula:                              result = -90.000
= ←GAGE2 - GAGE1, GAGE2.3 - GAGE1.3 )^
```

Look at the message box in the upper left corner of the screen.	The "Formula saved" message indicates that the CVIM system has saved the ATN formula in its memory. This completes the procedure for entering and saving the ATN formula.

Using Operator and Operand Keys (continued)

AVG Key

The AVG (average) function causes a list of operands to be added together and divided by the number of operands in the list.

If all operands in the list are *integers*, the result will also be an integer, rounded *down*. If *any* operand in the list is a fixed point number (such as 2.000), the result will also be a fixed point number, rounded to three places (such as 53.833).

The following procedure shows you how to set up a formula using the AVG operator. The procedure uses a list of six *integers*, then changes one of them to a *fixed point equivalent* to demonstrate the entry of an AVG formula.

Your Action	Comments
Select the second formula entry keyboard.	The AVG operator is on this keyboard.
Pick the AVG *key.*	When you pick the AVG key, the following entry appears in the upper box of the keyboard:

```
Formula:
= AVG(∧ ?
```

Note that the open parenthesis symbol, "(", appears with the AVG operator.

Pick these three keys: 8 9 *, (comma).*	This enters the first operand (89) in the list. The upper box should now appear as follows:

```
Formula:
= AVG ( 89, ∧ ?
```

Enter the remaining five operands, and insert a comma between each: 49,36,15, 20,114.	When you finish, the display should appear as follows:

```
Formula:
= AVG ( 89, 49, 36, 15, 20, 114 ∧ ?
```

Using Operator and Operand Keys (continued)

AVG Key (continued)

Your Action	Comments
Pick the "close parenthesis" key, ")".	This completes the AVG formula entry.

```
Formula:
= AVG ( 89, 49, 36, 15, 20, 114 )^
```

Pick the Ret *(return) key.* This causes the system to compute the average and display the "result" (53) in the box as follows:

```
Formula:                                    result = 53
= AVG ( 89, 49, 36, 15, 20, 114 )^
```

Note that since all operands in the list are *integers*, the result is *also* an integer; that is, the result is *truncated.*

Pick the left arrow (←) key. This causes the cursor (^) to move to the left one item. It should now be to the left of the close parenthesis. Note that the "result" has disappeared:

```
Formula:
= AVG ( 89, 49, 36, 15, 20, 114^ )
```

Pick the Del *(delete) key.* This deletes integer 114 and moves the cursor (^) to the right of the last comma:

```
Formula:
= AVG ( 89, 49, 36, 15, 20,^ ? )
```

Pick these five keys: 1 1 4 *. (decimal)* 0. This reenters 114 as a *fixed point number*: 114.0.

```
Formula:
= AVG ( 89, 49, 36, 15, 20, 114.0^ )
```

Using Operator and Operand Keys (continued) **AVG Key (continued)**

Your Action	Comments
Pick the Ret *(return) key.*	This causes the system to recompute the average and display the "result" (53.833) as a *fixed point number*. Note that the result has three digits to the right of the decimal point, and the number 114.0 now has two additional zeros:

```
Formula:                              result = 53.833
= AVG ( 89, 49, 36, 15, 20, 114.000^ )
```

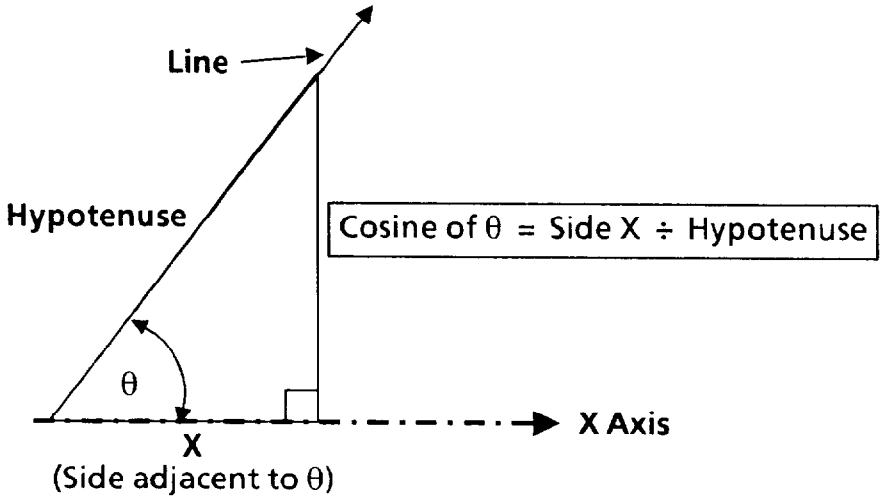
Look at the message box in the upper left corner of the screen.	The "Formula saved" message indicates that the CVIM system has saved the formula in its memory.

This completes the procedure for entering and saving the AVG formula.

COS Key

The COS (cosine) function computes the cosine of an angle (where the angle is stated in degrees).

The *cosine* of an unknown angle (in a right angle triangle) is the *ratio* between the side *adjacent* to the angle and the *hypotenuse*.



In a MATH-PAK formula, the COS operator requires one operand. The basic formula takes this form:

COS(angle in degrees)

The following procedure shows you how to set up a formula using the COS operator. The procedure uses a positive angle, stated in degrees, to demonstrate the entry of a COS formula.

Using Operator and Operand Keys *(continued)*

COS Key *(continued)*

Your Action	Comments
Select the second formula entry keyboard.	The COS operator is on this keyboard.
Pick the COS *key.*	When you pick the COS key, the following entry appears in the upper box of the keyboard:

```
Formula:
= COS(^ ?
```

Pick these two keys: 6 0.	This enters the value of the positive angle in degrees. The display should now appear as follows:

```
Formula:
= COS ( 60^ ?
```

Pick the "close parenthesis" key, ")".	This completes the COS formula entry.

```
Formula:
= COS ( 60 )^
```

Pick the Ret *(return) key.*	This causes the system to display the "result." In this case, the result is 0.500, which is the cosine of an angle of 60°.

```
Formula:                                result = 0.500
= COS ( 60 )^
```

Look at the message box in the upper left corner of the screen.	This indicates that the CVIM system has saved the COS formula in its memory. This completes the procedure for entering and saving the COS formula.

Using Operator and Operand Keys (continued)

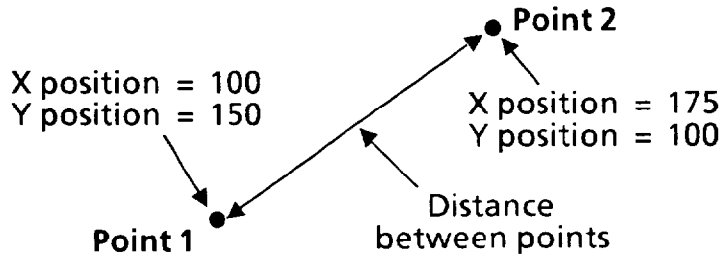
DST Key

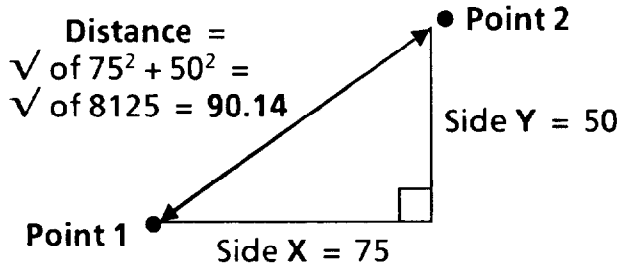
The DST (distance) function computes the distance from one position in the image field to another position.

It performs the distance computation by using the Pythagorean theorem: *The square of the hypotenuse of a right angle triangle is equal to the sum of the squares of the two sides.*

Thus, to find the distance between two points in the image field (the *hypotenuse* of a right angle triangle), the DST function squares the distance along the X axis (*one side* of the triangle) and the Y axis (the *other side* of the triangle), then computes the square root of the sum of the two squares.

For example, if the *first* point has an X position of 100 pixels and a Y position of 150 pixels, and the *second* point has an X position of 175 pixels and a Y position of 100, the distance between these points is equal to the square root of the sum of the squared distance along the X axis and the squared distance along the Y axis, as follows:





In a MATH-PAK formula, the DST operator requires four operands. The basic formula takes this form:

$$DST(X_1, Y_1, X_2, Y_2)$$

In the formula, operands X_1 and Y_1 are the coordinates of Point 1 in the illustration above; operands X_2 and Y_2 are the coordinates of Point 2.

The following procedure shows you how to set up a formula using the DST operator. The procedure follows the example above to demonstrate the entry of a DST formula.

Using Operator and Operand Keys *(continued)*

DST Key *(continued)*

Your Action	Comments
Select the second formula entry keyboard.	The DST operator is on this keyboard.
Pick the DST *key.*	When you pick the DST key, the following entry appears in the upper box of the keyboard:

```
Formula:
= DST(^ ?
```

Pick these four keys: 1 0 0 *, (comma).*	This enters the X position of the first point in the image field. The display should now appear as follows:

```
Formula:
= DST ( 100^ ?
```

Enter the values for Y_1, X_2 *and* Y_2 *with commas between them:* 150,175,100.	This enters the Y position of the first point and the X and Y positions of the second point. The display should now appear as follows:

```
Formula:
= DST ( 100, 150, 175, 100^ ?
```

Pick the "close parenthesis" key, ")".	This completes the DST formula entry.

```
Formula:
= DST ( 100, 150, 175, 100 )^
```

Pick the Ret *(return) key.*	This causes the system to display the "result." In this case, the result is 90.138, which is the distance (in pixels) between the two points.

```
Formula:                                result = 90.138
= DST ( 100, 150, 175, 100 )^
```

Using Operator and Operand Keys (continued)

DST Key (continued)

Your Action	Comments
Look at the message box in the upper left corner of the screen.	The "Formula saved" message indicates that the CVIM system has saved the DST formula in its memory. This completes the procedure for entering and saving the DST formula.

MAX, MIN Keys

The MAX (maximum) and MIN (minimum) functions find the largest and smallest values, respectively, in a list of operands.

In a MATH-PAK formula, the MAX and MIN operators require two or more operands. The basic formulas take this form:

MAX(list) and MIN(list)

The following procedure shows you how to set up a formula using the MAX or MIN operator. The procedure uses a list of integers to demonstrate the entry of these formulas.

Your Action	Comments
Select the second formula entry keyboard.	The MAX and MIN operators are on this keyboard.
Pick the MAX *key.*	When you pick the MAX key, the following entry appears in the upper box of the keyboard: Formula: = MAX(^ ?
Pick these three keys: 1 2 *,(comma).*	This enters the first operand in the list, along with a comma. The display should now appear as follows: Formula: = MAX(12,^ ?
Select the first formula entry keyboard.	The "unary" (±) operator is on this keyboard.

Using Operator and Operand Keys *(continued)*

MAX, MIN Keys *(continued)*

Your Action	Comments
Enter three more operands: 5,89,–100.	This enters the remaining operands in the list. Use the unary key (±) to enter the minus (–) sign ahead of the "100." The display should now appear as follows:

```
Formula:
= MAX ( 12, 5, 89, - 100^ ?
```

Select the second formula entry keyboard.	The parenthesis keys are on this keyboard.
Pick the "close parenthesis" key, ")".	This completes the MAX formula entry.

```
Formula:
= MAX ( 12, 5, 89, - 100 )^
```

Pick the Ret *(return) key.*	This causes the system to display the "result." In this case, the result is 89, which is the maximum value in the list.

```
Formula:                                  result = 89
= MAX ( 12, 5, 89, - 100 )^
```

Look at the message box in the upper left corner of the screen.	This indicates that the CVIM system has saved the formula in its memory. This completes the procedure for entering and saving the MAX formula. Now, use the following steps to change to the MIN operator and demonstrate its operation.
Pick the left arrow (←) key repeatedly until the cursor is to the right of MAX (.	This causes the cursor (^) to move to the *left*. The display should appear as follows:

```
Formula:
= MAX (^ 12, 5, 89, - 100 )
```

Using Operator and Operand Keys *(continued)* — **MAX, MIN Keys** *(continued)*

Your Action	Comments
Pick the Del *(delete) key.*	This deletes the MAX operator and moves the cursor (∧) to the left as follows:

```
Formula:
= ∧ 12 ?, 5, 89, - 100 )
```

Pick the MIN *key.*	When you pick the MIN key, the MIN operator appears. The display should now appear as follows:

```
Formula:
= MIN (∧ 12, 5, 89, - 100 )
```

Pick the Ret *(return) key.*	This causes the system to display the "result." In this case, the result is –100, which is the minimum value in the list.

```
Formula:                                result = -100
= MIN (∧ 12, 5, 89, - 100 )
```

Look at the message box in the upper left corner of the screen.	The "Formula saved" message indicates that the CVIM system has saved the formula in its memory. This completes the procedure for entering and saving the MAX (or MIN) formula.

MED Key

The MED (median) function sorts a list of operands in ascending order, then returns the value at the *center* position for an *odd* number of operands, or the *higher order* of the two center values for an *even* number of operands.

For example, in a list with an *odd number* of operands . . .

82, 3, 86, 2, 43

. . . the MED function first sorts their values in ascending order . . .

2, 3, 43, 82, 86

. . . then returns the *center* value, 43.

Using Operator and Operand Keys (continued)

MED Key (continued)

In a list with an *even number* of operands . . .

82, 3, 86, 2, –15, 8

. . . the MED function first sorts their values in ascending order . . .

–15,2, 3, 8, 82, 86

. . . then returns the *high order value* of the two *center* values. In the preceding list, the two center values are 3 and 8; thus, the high order value is 8.

The basic formula takes this form:

MED(list)

The following procedure shows you how to set up a formula using the MED operator. The procedure uses a list of six integers to demonstrate the entry of a MED formula.

Your Action	Comments
Select the second formula entry keyboard.	The MED operator is on this keyboard.
Pick the MED *key.*	When you pick the MED key, the following entry appears in the upper box of the keyboard:

```
Formula:
= MED(∧ ?
```

Pick these three keys: 8 2 *,(comma).*	The display should now appear as follows:

```
Formula:
= MED ( 82, ∧ ?
```

Select the first formula entry keyboard.	The "unary" operator (±) is on this keyboard.
Pick the (±) key.	When you pick the ± key, the following entry appears in the display:

```
Formula:
= MED( 82, –∧ ?
```

This will *negate* the value that follows it.

Using Operator and Operand Keys (continued)

MED Key (continued)

Enter the remaining operands (with a comma between each): 15,3,86,2,8.

The display should now appear as follows:

```
Formula:
= MED ( 82, – 15, 3, 86, 2, 8^ ?
```

Select the second formula entry keyboard.

The parenthesis keys are on this keyboard.

Pick the "close parenthesis" key, ")".

This completes the MED formula entry.

```
Formula:
= MED ( 82, – 15, 3, 86, 2, 8 )^
```

Pick the Ret *(return) key.*

This causes the system to display the "result." In this case, the result is 8, which is the "median" value in this list.

```
Formula:                                  result = 8
= MED ( 82, – 15, 3, 86, 2, 8 )^
```

Look at the message box in the upper left corner of the screen.

The "Formula saved" message indicates that the CVIM system has saved the formula in its memory.

This completes the procedure for entering and saving the MED formula.

MOD Key

The MOD (mode) function finds the value in a list of operands that appears most often. If no value appears any more often than any other value, the MOD function returns the *lowest* value in the list.

For example, in the following list of operands . . .

2, 3, 2, 3, 3

. . . the MOD function returns 3, since it occurs more often than 2.

The basic formula takes this form:

MOD(list)

Using Operator and Operand Keys (continued)

MOD Key (continued)

The following procedure shows you how to set up a formula using the MOD operator. The procedure uses the preceding list of five integers to demonstrate the entry of a MOD formula.

Your Action	Comments
Select the second formula entry keyboard.	The MOD operator is on this keyboard.
Pick the MOD *key.*	When you pick the MOD key, the following entry appears in the upper box of the keyboard:

```
Formula:
= MOD(^ ?
```

Pick these two keys: 2 *,(comma).*	The display should now appear as follows:

```
Formula:
= MOD ( 2, ^ ?
```

Enter the remaining operands (with a comma between each): 3,2,3,3.	The display should now appear as follows:

```
Formula:
= MOD ( 2, 3, 2, 3, 3^ ?
```

Pick the "close parenthesis" key, ")".	This completes the MOD formula entry.

```
Formula:
= MOD ( 2, 3, 2, 3, 3 )^
```

Pick the Ret *(return) key.*	This causes the system to display the "result." In this case, the result is 3, which is the "mode" value in this list.

```
Formula:                                  result = 3
= MOD ( 2, 3, 2, 3, 3 )^
```

Using Operator and Operand Keys (continued)

MOD Key (continued)

Look at the message box in the upper left corner of the screen.

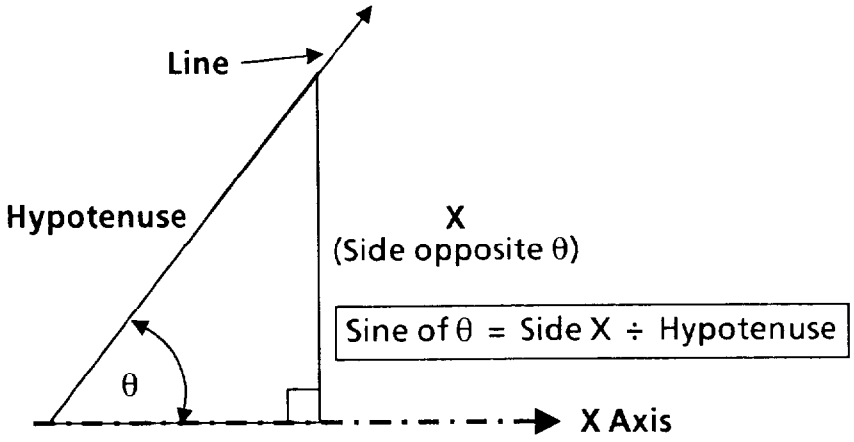
The "Formula saved" message indicates that the CVIM system has saved the formula in its memory.

This completes the procedure for entering and saving the MOD formula.

SIN Key

The SIN (sine) function computes the sine of an angle (where the angle is stated in degrees).

The *sine* of an unknown angle (in a right angle triangle) is the *ratio* between the side *opposite* the angle and the *hypotenuse.*



In a MATH-PAK formula, the SIN operator requires one operand. The basic formula takes this form:

SIN(angle in degrees)

The following procedure shows you how to set up a formula using the SIN operator. The procedure uses a positive angle, stated in degrees, to demonstrate the entry of a SIN formula.

Your Action	Comments
Select the second formula entry keyboard.	The SIN operator is on this keyboard.
Pick the SIN *key.*	When you pick the SIN key, the following entry appears in the upper box of the keyboard:

```
Formula:
= SIN(∧ ?
```

Using Operator and Operand Keys *(continued)*

SIN Key *(continued)*

Your Action	Comments
Pick these two keys: 3 0.	This enters the value of the positive angle in degrees. The display should now appear as follows:

```
Formula:
= SIN ( 30^ ?
```

Pick the "close parenthesis" key, ")".	This completes the SIN formula entry.

```
Formula:
= SIN ( 30 )^
```

Pick the Ret *(return) key.*	This causes the system to display the "result." In this case, the result is 0.500, which is the sine of an angle of 30°.

```
Formula:                          result = 0.500
= SIN ( 30 )^
```

Look at the message box in the upper left corner of the screen.	The "Formula saved" message indicates that the CVIM system has saved the formula in its memory.

This completes the procedure for entering and saving the SIN formula.

SQR Key

The SQR (square root) function computes the square root of a value. It always returns a *fixed point* number, regardless of whether the original value was an integer or a fixed point number.

For example, if the value entered is 49, the SQR function returns 7.000.

In a MATH-PAK formula, the SQR operator requires one operand. The basic formula takes this form:

SQR(value)

The following procedure shows you how to set up a formula using the SQR operator.

Using Operator and Operand Keys *(continued)*

SQR Key *(continued)*

Your Action	Comments
Select the second formula entry keyboard.	The SQR operator is on this keyboard.
Pick the SQR *key.*	When you pick the SQR key, the following entry appears in the upper box of the keyboard:

```
Formula:
= SQR(^ ?
```

Pick these two keys: 4 8.	The display should now appear as follows:

```
Formula:
= SQR ( 48, ^ ?
```

Pick the "close parenthesis" key, ")".	This completes the SQR formula entry.

```
Formula:
= SQR ( 48 )^
```

Pick the Ret *(return) key.*	This causes the system to display the "result." In this case, the result is 6.927, which is the square root of 48.

```
Formula:                          result = 6.927
= SQR ( 48 )^
```

Look at the message box in the upper left corner of the screen.	The "Formula saved" message indicates that the CVIM system has saved the formula in its memory. This completes the procedure for entering and saving the SQR formula.

Using Operator and Operand Keys (continued)

SQ Key

The SQ (square) function computes the square of a value. It returns an integer if the value squared is an integer, and returns a *fixed point* number if the value squared is a fixed point number.

For example, if the value to be squared is 7, the SQ function returns 49. If the value to be squared is 7.000, the SQ function returns 49.000.

In a MATH-PAK formula, the SQ operator requires one operand. The basic formula takes this form:

SQ(value)

The following procedure shows you how to set up a formula using the SQ operator.

Your Action	Comments
Select the second formula entry keyboard.	The SQ operator is on this keyboard.
Pick the SQ *key.*	When you pick the SQ key, the following entry appears in the upper box of the keyboard:

```
Formula:
= SQ(^ ?
```

Pick these two keys: 1 2.	The display should now appear as follows:

```
Formula:
= SQ ( 12 ^ ?
```

Pick the "close parenthesis" key, ")".	This completes the SQ formula entry.

```
Formula:
= SQ ( 12 )^
```

Pick the Ret *(return) key.*	This causes the system to display the "result." In this case, the result is 144, which is the square of 12.

```
Formula:                                result = 144
= SQ ( 12 )^
```

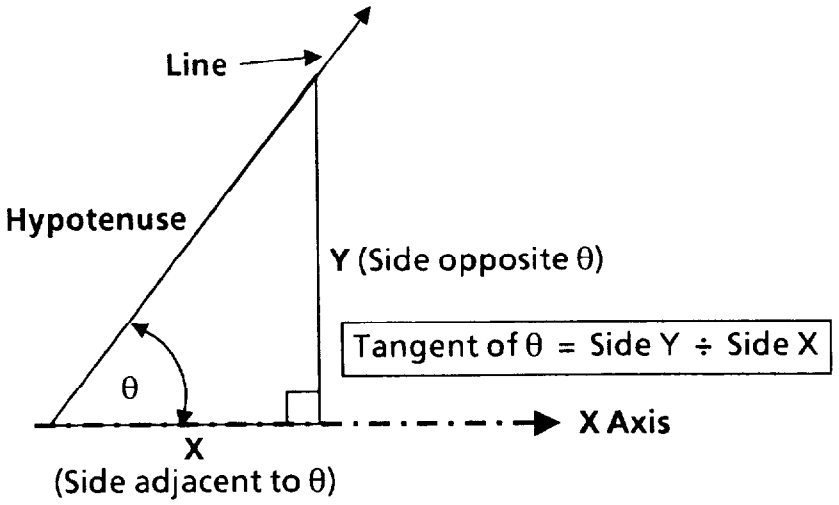
SQ Key (continued)

Your Action	Comments
Look at the message box in the upper left corner of the screen.	The "Formula saved" message indicates that the CVIM system has saved the formula in its memory.
	This completes the procedure for entering and saving the SQ formula.

TAN Key

The TAN (tangent) function computes the tangent of an angle (where the angle is stated in degrees).

The *tangent* of an unknown angle (in a right angle triangle) is the *ratio* between the side *opposite* the angle and the side *adjacent* to the angle.



In a MATH-PAK formula, the TAN operator requires one operand. The basic formula takes this form:

TAN(angle in degrees)

Using Operator and Operand Keys *(continued)*

TAN Key *(continued)*

The following procedure shows you how to set up a formula using the TAN operator. The procedure uses a positive angle, stated in degrees, to demonstrate the entry of a TAN formula.

Your Action	Comments
Select the second formula entry keyboard.	The TAN operator is on this keyboard.
Pick the TAN *key.*	When you pick the TAN key, the following entry appears in the upper box of the keyboard:

```
Formula:
= TAN(^ ?
```

Pick these two keys: 4 5.	This enters the value of the positive angle in degrees. The display should now appear as follows:

```
Formula:
= TAN ( 45^ ?
```

Pick the "close parenthesis" key, ")".	This completes the TAN formula entry.

```
Formula:
= TAN ( 45 )^
```

Pick the Ret *(return) key.*	This causes the system to display the "result." In this case, the result is 1.000, which is the tangent of an angle of 45°.

```
Formula:                          result = 1.000
= TAN ( 45 )^
```

Look at the message box in the upper left corner of the screen.	The "Formula saved" message indicates that the CVIM system has saved the formula in its memory.
	This completes the procedure for entering and saving the TAN formula.

Using Operator and Operand Keys (continued)

VAR Key

The VAR (variance) function computes the variance of a list of operands. The variance is equal to the sum of the square of the difference between the value of each operand and the average value of all operands, divided by the number of operands in the list.

Using the following list of operands . . .

5, 8, 10

. . . the VAR function processes the list as follows:

1. It computes the *average* value of all operands. Thus, 5 + 8 + 10 = 23, and 23 ÷ 3 = 7.667, the average value of the three operands.

2. It computes the difference between the value of each operand and the list average, 7.667, then computes the square of each difference, as follows:

5 – 7.667 = –2.667; –2.667 × –2.667 = 7.113

8 – 7.667 = 0.333; 0.333 × 0.333 = 0.111

10 – 7.667 = 2.333; 2.333 × 2.333 = 5.443

3. It computes the *average* of the squared differences, as follows:

7.113 + 0.111 + 5.443 = 12.667; 12.667 ÷ 3 = 4.222

Thus, the *variance* is 4.222.

In a MATH-PAK formula, the VAR operator requires two or more operands. The basic formula takes this form:

VAR(list)

The following procedure shows you how to set up a formula using the VAR operator. The procedure uses the preceding list of integers to demonstrate the entry of a VAR formula.

Select the second formula entry keyboard.

The VAR operator is on this keyboard.

Pick the VAR *key.*

When you pick the VAR key, the following entry appears in the upper box of the keyboard:

```
Formula:
= VAR(^ ?
```

Using Operator and Operand Keys (continued)

VAR Key (continued)

Your Action	Comments
Pick these two keys: 5 , *(comma).*	This enters the first value in the list, along with a comma. The display should now appear as follows:

```
Formula:
= VAR( 5,^ ?
```

Enter the remaining operands (with a comma between each): 8,10.	The display should now appear as follows:

```
Formula:
= VAR ( 5, 8, 10^ ?
```

Pick the "close parenthesis" key, ")".	This completes the VAR formula entry.

```
Formula:
= VAR ( 5, 8, 10 )^
```

Pick the Ret *(return) key.*	This causes the system to display the "result." In this case, the result is 4.221, which is the variance value for this list.

```
Formula:                                result = 4.221
= VAR ( 5, 8, 10 )^
```

Because of rounding errors that sometimes occur with fixed point calculations, the result is 4.221 instead of 4.222, as shown earlier. Normally, this will not affect the usefulness of these calculations.

Look at the message box in the upper left corner of the screen.	The "Formula saved" message indicates that the CVIM system has saved the formula in its memory.

This completes the procedure for entering and saving the VAR formula.

Using Operator and Operand Keys (continued)

Arithmetic Operator Keys: +, –, *, /

The arithmetic operators are used between two values to perform addition, subtraction, multiplication, and division functions, as shown in the following examples.

Addition: By placing the *addition* operator (+) between the following integers . . .

52 + 34

. . . the system adds the two integers and computes a result of 86.

Subtraction: By placing the *subtraction* operator (–) between the following integers . . .

52 – 34

. . . the system subtracts the *right* integer from the *left* integer and computes a result of 18.

Multiplication: By placing the *multiplication* operator (*) between the following integers . . .

52 * 34

. . . the system multiplies the two integers and computes a result of 1768.

Division: By placing the *division* operator (/) between the following two integers . . .

52 / 34

. . . the system divides the *left* integer by the *right* integer and computes a result of 1 (the result is truncated). (If one of the two numbers were *fixed point*, the result would *also* be fixed point: 1.529.)

Parenthesis Keys: ( )

The parenthesis are used to enclose an arithmetic or logical expression within a formula. This ensures that the formula will evaluate the *contents* of the expression as an *entity*.

For example, in the following formula . . .

3 + 6 * 5

. . . the formula evaluates "6 * 5" *first*; thus, in this case the result is 33 (6 * 5 = 30, and 30 + 3 = 33).

By placing parenthesis around "3 + 6" . . .

(3 + 6) * 5

. . . the formula evaluates it as a single entity (3 + 6 = 9); thus, in this case the result is 45 (9 * 5 = 45).

Using Operator and Operand Keys (continued)

Modulo Operator Key: %

The modulo operator is used between two values to return a remainder, if any resulting from the division operation.

For example, by placing the modulo operator (%) between the following two integers . . .

52 % 34

. . . the system divides the *left* integer by the *right* integer, then returns the *remainder* from the division operation. Since 34 goes into 52 once, the remainder is 52 – 34 = 18. Thus, the system returns a result of 18.

Unary Operator Key: ±

The unary operator can be entered just ahead of an operand in a formula in order to *negate* the value of the operand.

For example, when entering the following list of integers . . .

15,–27,133,6

. . . press the unary key (±) to place a minus sign (–) before entering the integer 27. This causes 27 to become a *negative* integer.

Comma Key: ,

The comma is used to separate the operands in a list, such as in an AVG formula:

AVG(21,54,101,3)

Logic Operator Keys

The logic operators can perform logical operations when placed between two values in a formula. The result from these operations is either a logic "1" or logic "0," according to whether or not the logic condition is satisfied for the particular logic operator and values used.

The following examples show how the logic operators work.

AND: Assume that the inspection result from GAGE1 is 6 and the result from GAGE2 is 3. By placing the "AND" operator between GAGE1 and GAGE2 . . .

GAGE1 AND GAGE2

. . . the formula will return a logic "1" result, since *both* values are nonzero.

If *either* value were zero, or if *both* values were zero, the formula would return a logic "0" result.

Using Operator and Operand Keys (continued)

Logic Operator Keys (continued)

OR: Assume that the inspection result from GAGE1 is 0 and the result from GAGE2 is 3. If you place the "OR" operator between GAGE1 and GAGE2 . . .

GAGE1 OR GAGE2

. . . the formula will return a logic "1" result, since at least *one* value is nonzero.

If *both* values were zero, the formula would return a logic "0" result.

NOT: The "not" operator (NOT) is used with expressions containing other logic operators. Its effect is to *invert* the normal logic result. For example, if the normal result of an "AND" operation were a logic "1," the "NOT" operator will invert it to a logic "0."

Assume that the inspection result from WIN1 is 400 and the result from WIN2 is 500. Assume further that these tools are "ANDed" in a formula. If you place the "NOT" operator ahead of the "AND" expression . . .

NOT (WIN1 AND WIN2)

. . . the formula will return a logic "0." (In this case, since *both* values are nonzero, the *normal* "AND" operation result is a logic "1"; however, the "NOT" operator inverts the result to a logic "0.")

=: The "equal" operator (=) returns a logic "1" when placed between two operands whose results are *equal.* Thus, if the GAGE1 and GAGE2 results are both 6 in the following formula . . .

GAGE1 = GAGE2

. . . the formula will return a logic "1" result, since the two gage results are *equal.*

If the two gage results were *not equal,* the formula would return a logic "0."

<>: The "not equal" operator (<>) returns a logic "1" when placed between two operands whose results are *not equal.* Thus, if the GAGE1 result is 5 and the GAGE2 result is 3 in the following formula . . .

GAGE1 <> GAGE2

. . . the formula will return a logic "1" result, since the two gage results are *not equal.*

If the two gage results were *equal,* the formula would return a logic "0."

Using Operator and Operand Keys *(continued)*

Logic Operator Keys *(continued)*

>: The "greater than" operator (>) returns a logic "1" when placed between two operands where the left operand result is *greater than* the right operand result. Thus, if the GAGE1 result is 8 and the GAGE2 result is 7 in the following formula . . .

GAGE1 > GAGE2

. . . the formula will return a logic "1" result, since 8 is greater than 7.

If the GAGE1 result were equal to, or less than, the GAGE2 result, the formula would return a logic "0."

<: The "less than" operator (<) returns a logic "1" when placed between two operands where the left operand result is *less than* the right operand result. Thus, if the GAGE1 result is 7 and the GAGE2 result is 8 in the following formula . . .

GAGE1 < GAGE2

. . . the formula will return a logic "1" result, since 7 is greater than 8.

If the GAGE1 result were equal to, or greater than, the GAGE2 result, the formula would return a logic "0."

>=: The "greater than or equal to" operator (>=) returns a logic "1" when placed between two operands where the left operand result is *greater than, or equal to*, the right operand result. Thus, if the GAGE1 result is 46 and the GAGE2 result is 45 in the following formula . . .

GAGE1 >= GAGE2

. . . the formula will return a logic "1" result, since 46 is greater than 45. The result would be the same, in this case, if the two gage results were both 45 or 46

If the GAGE1 result were less than the GAGE2 result, the formula would return a logic "0."

<=: The "less than or equal to" operator (<=) returns a logic "1" when placed between two operands where the left operand result is *less than, or equal to*, the right operand result. Thus, if the GAGE1 result is 32 and the GAGE2 result is 33 in the following formula . . .

GAGE1 >= GAGE2

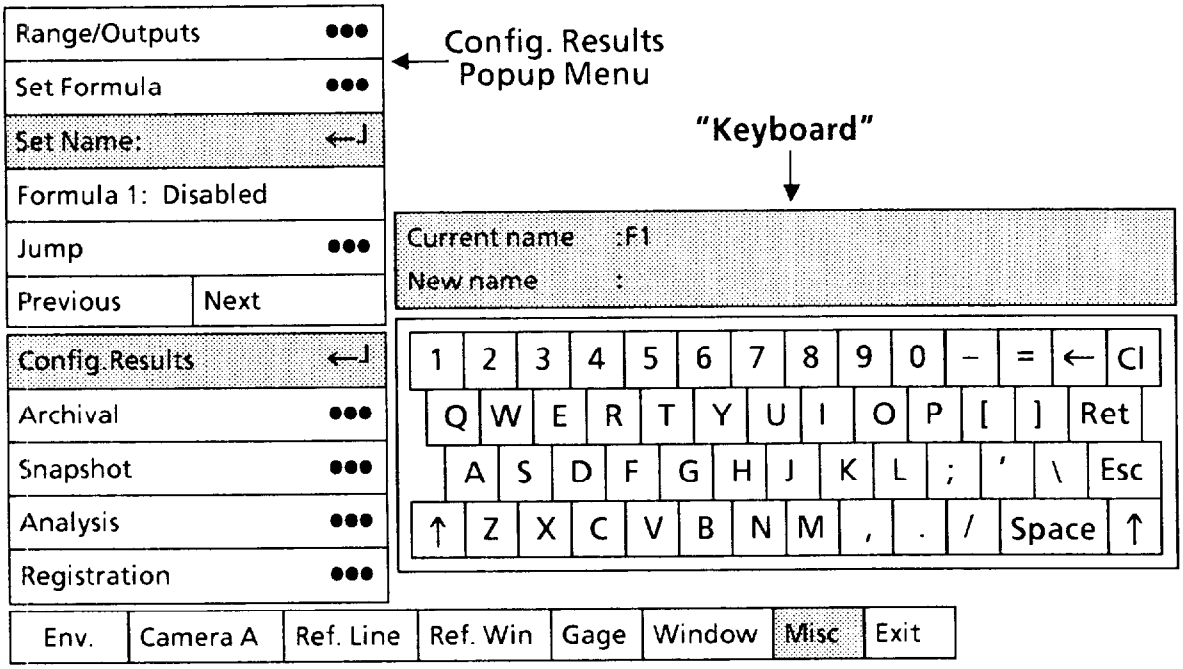
. . . the formula will return a logic "1" result, since 32 is less than 33. The result would be the same, in this case, if the two gage results were both 32 or 33

If the GAGE1 result were greater than the GAGE2 result, the formula would return a logic "0."

Setting Formula Name

Use this function to name the current formula. You can choose any name consisting of up to seven alphanumeric, punctuation, and miscellaneous printable characters, including the "space" character.

Your Action	Comments
Pick the Set Names *menu box, if appropriate.*	When you pick the Set Name menu box, a "keyboard" appears alongside the Config. Results popup menu, as follows:



The default name is always the number of the current formula. In the example above, it is F1.

The keyboard has five function keys:

1. The *backspace* key (←) deletes the last character entered and moves the cursor (^) left one space.

2. The Esc (*"escape"*) key returns to the Set Archive Names popup menu without changing the current name, if any.

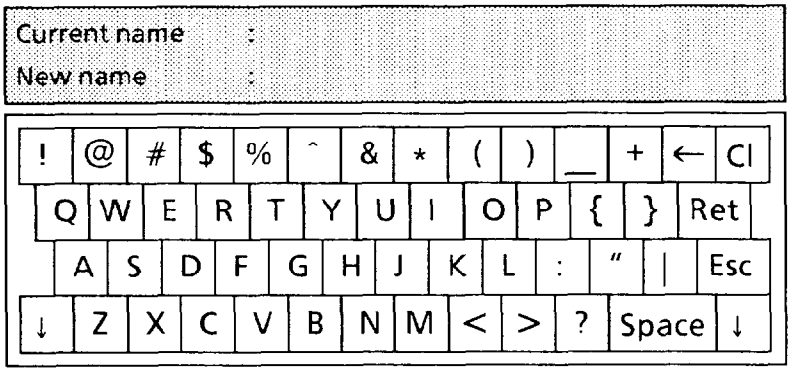
3. The Cl (*"clear"*) key sets the "New name" line to all blanks.

4. The Ret (*"return"*) key enters the new name into the appropriate menu box in the Set Archive Names popup menu.

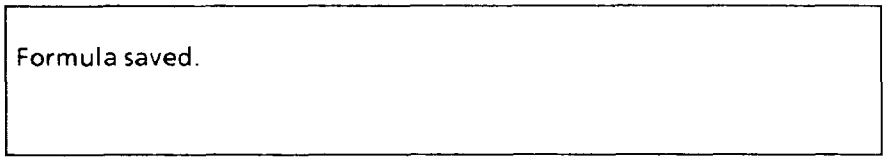
5. The *shift* keys (↑ ↓) select alternate "keyboards" by causing several of the keyboard keys to change. The *alphabet* characters, however, remain the same on both keyboards.

Setting Formula Name *(continued)*

Your Action	Comments
	The preceding keyboard illustration shows the *first* keyboard, while the following illustration shows the *second* keyboard.



Pick each character of the new name.	As you pick the characters, they will appear after New name.
Pick the Ret *key to enter the new formula name.*	When you pick the Ret key, the new name will appear in the Set Name menu box in the Config. Results menu.
	Note also that the "Formula saved" message appears in the message box on the monitor screen:



This indicates that the name has been saved in memory.

Enabling Formula

Use this function to *enable* the currently selected formula.

NOTE: When you enable a formula, the CVIM system prevents an associated gage or window from sending its results to the discrete outputs.

For example, if you enable formula #1, the results of its operation will be available to the discrete outputs *instead of* the results from a window #1 inspection. The CVIM system will use the range limits and discrete outputs assigned to formula #1 and send the results of *its* operation to those outputs. (Window #1 results data are still available through the other I/O ports, however.)

Enabling Formula *(continued)*

The following table relates each formula number to the number of the gage or window whose results it replaces.

Formula	Tool	Formula	Tool	Formula	Tool	Formula	Tool
1	Window 1	15	Gage 7	29	Gage 13	43	Gage 19
2	Window 2	16	Gage 8	30	Gage 14	44	Gage 20
3	Window 3	17	Window 9	31	Gage 15	45	Gage 21
4	Window 4	18	Window 10	32	Gage 16	46	Gage 22
5	Window 5	19	Window 11	33	Window 17	47	Gage 23
6	Window 6	20	Window 12	34	Window 18	48	Gage 24
7	Window 7	21	Window 13	35	Window 19	49	Gage 25
8	Window 8	22	Window 14	36	Window 20	50	Gage 26
9	Gage 1	23	Window 15	37	Window 21	51	Gage 27
10	Gage 2	24	Window 16	38	Window 22	52	Gage 28
11	Gage 3	25	Gage 9	39	Window 23	53	Gage 29
12	Gage 4	26	Gage 10	40	Window 24	54	Gage 30
13	Gage 5	27	Gage 11	41	Gage 17	55	Gage 31
14	Gage 6	28	Gage 12	42	Gage 18	56	Gage 32

Your Action	Comments
Pick the Formula XX *menu box.*	When you pick the Formula XX menu box ("XX" is the formula number), a warning message appears in the message box, as follows:

> WARNING: Enabling this formula will
> prevent Window #1 from being range
> checked or assigned to outputs.
> Reselect to confirm.

Note "Window #1" in the box. The *actual* tool and number depend on which formula number is currently selected.

Pick the Formula XX *menu box again, if appropriate.*

When you pick the Formula XX menu box again, the warning message disappears, and "Enabled" replaces "Disabled" in the menu box.

NOTE: If you decide to *disable* a currently enabled formula, another warning message will appear when you pick the Formula XX menu box, as follows:

> WARNING: Disabling this formula will
> allow Window #1 to be range checked.
> The range for this window must be enabled.
> Reselect to confirm.

To disable the formula, you must pick the Formula XX menu box a second time.

Assigning Range Limits and Outputs

Range Limits – the term *range limit*, as it applies to a MATH-PAK formula, refers to a *formula result* above or below which the CVIM system might not (or cannot) successfully perform an inspection.

In general, range limits specify the upper and lower boundaries of acceptable inspection results.

The CVIM system provides *two* sets of range limits: *warning range* limits, and *fault range* limits. Warning range limits must always lie at or *within* fault range limits.

The two sets of range limits have this relationship:

LF< =LW< =NOMINAL< =UW< =UF

The **NOMINAL** value refers to a desired middle-of-the-range value. **LW** and **UW** are the lower and upper warning limits. **LF** and **UF** are the lower and upper fault limits.

Here is a general example, using a nominal value of 100, to demonstrate the concept:

- The **LW** value (lower warning limit) could be set to 95.
- The **UW** value (upper warning limit) could be set to 105.
- The **LF** value (lower fault limit) could be set to 90.
- The **UF** value (upper fault limit) could be set to 110.

During an inspection, if the formula result goes outside either warning limit, the CVIM system will generate a warning signal. If the formula result goes outside *both* a warning limit *and* a fault limit, the CVIM system will generate a warning signal *and* a fault signal.

In a practical application, a warning signal can function as an early indication of a deteriorating condition. A fault signal can indicate a "hard" failure, such as a broken part. In each case, the signal that results when a range limit is exceeded can be used to alert operations personnel to fix the problem.

Outputs – the term *outputs* refers to the 14 discrete outputs that you can configure to carry various signals to your production equipment. Of these signals, the "results" signals indicate whether or not any of the warning and/or fault range limits have been exceeded.

The *CVIM User's Manual* (Catalog No. 5370-ND001), Chapter 4, *Operating Environment*, shows you how to assign signal functions to the discrete outputs.

In this section, you can select warning and fault limits for your formula results and assign them to any of the discrete outputs that you designated earlier for "results" outputs.

Assigning Range Limits and Outputs (continued)

The following steps describe a procedure for performing a trial series of inspections using a representative sample of workpieces in order to accumulate a statistical basis for setting the range limits for the formulas.

Your Action	Comments
Prepare to run a series of "trial" inspections.	Refer to the *CVIM User's Manual*, Chapter 10, *Runtime Operations*, for more details about the following steps. For these trial inspection series, you should have on hand a sufficiently large quantity of representative workpieces. If you cannot use the actual factory-floor setting to perform these trial inspections, you can manually position each workpiece in front of the camera and use a manual trigger, or use some type of automatic positioning and triggering mechanism that approximates the factory-floor situation.
Pick the Exit *menu box in the Main Configuration menu.*	When you pick the Exit menu box, the Exit popup menu will appear.
Pick the Runtime Display *menu box in the* Exit *menu.*	When you pick the Runtime Display menu box, the Runtime Display popup menu will appear.
Pick the Stat. Page 1 *menu box in the* Runtime Display *menu.*	This causes the Stat. Page 1 statistics table to appear on the monitor screen when you activate the run mode. Stat. Page 1 can be configured to display "results" statistics for all *enabled* formulas, gages, and windows.
Pick the Runtime Init. *menu box in the* Exit *menu.*	When you pick the Runtime Init. menu box, the Runtime Init. popup menu appears.
Look at the Mode *menu box in the* Runtime Init. *menu.*	If "Standard" appears, pick the box *once* to toggle to "Learn." This activates the "learn" mode during the trial inspection series and ensures the accumulation of "results" data for this formula in the Stat. Page 1 table.
Look at the Disp. Formulas *menu box in the* Runtime Init.*menu.*	If "0" appears, pick the box to display the "calculator" pad. Enter "1" (or higher) in order to display statistics from at least *one* formula (per page) in the Stat Page 1 table. For example, if you enter "1" and you want to see the statistics from formula #5, after activating the run mode you must "page down" the Stat Page 1 table *four times* in order to see the formula #5 statistics.
Look at the Stat. Samples *menu box in the* Runtime Init. *menu.*	This menu box enables you to specify the number of inspections that you want the CVIM system to perform before it displays accumulated statistical data in the Stat Page 1 table.

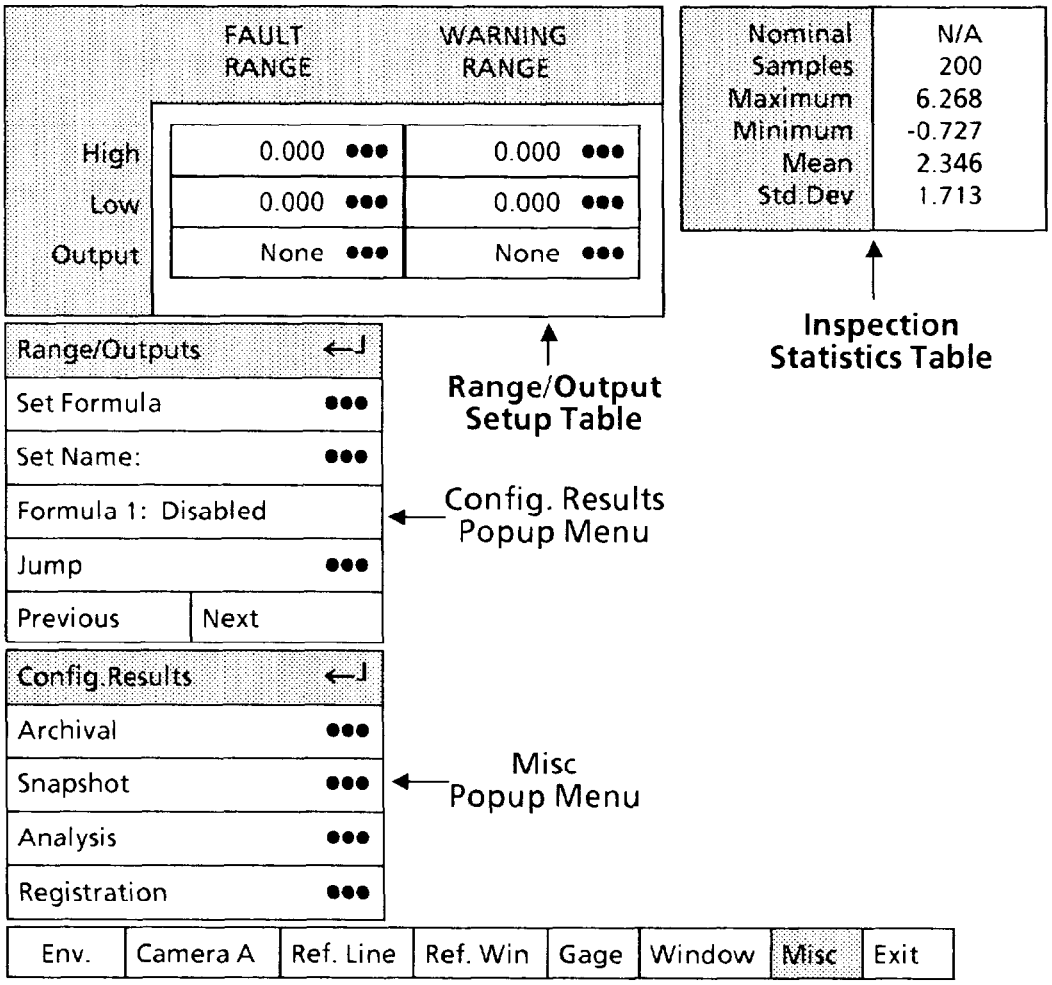
Assigning Range Limits and Outputs (continued)

Your Action	Comments
Pick the Stat. Samples *menu box, if appropriate.*	When you pick the box, the "calculator" pad appears. Enter the number of inspections to be performed for each sample.
Pick the Runtime *menu box in the* Exit *menu.*	When you pick the Runtime menu box, the Runtime popup menu appears.
Pick the Goto Runmode *menu box in the* Runtime *menu.*	When you pick the Goto Runmode menu box, the CVIM system begins running inspections *if you selected Auto/Internal as the trigger source.* If not, the system will await trigger inputs from whatever trigger source you selected.
Look the Stat. Page 1 *table.*	As the inspections continue, the Stat. Page 1 table displays accumulated "results" data for each *enabled* gage, window, and/or formula.
When the desired number of inspections appears under "samples" in the Stat. Page 1 *table, pick the* Setup *menu box to stop the run mode.*	Picking the Setup menu box *stops* the run mode and returns the CVIM system to the configuration mode. At this time, the *final* data appearing in the Stat. Page 1 table are recorded in the Inspection Statistics table for the currently selected formula. You will see this when you pick the Range/Outputs menu box for the appropriate formula.
Look at the four statistics columns in the Stat. Page 1 *table.*	These columns show the *mean, standard deviation, minimum reading,* and *maximum reading* statistics for the trial inspection series. These statistics are your basis for configuring the range limit values for the currently selected formula.
Pick the Misc *menu box.*	This restores the Misc popup menu.
Pick the Config. Results *menu box.*	This restores the Config. Results popup menu.

Assigning Range Limits and Outputs *(continued)*

Use the following steps to configure the appropriate range limits for your MATH-PAK formulas.

Your Action	Comments
Pick the Range/Outputs *menu box in the* Config. Results *popup menu.*	When you pick the Range/Outputs menu box, two tables will appear on the screen, as follows:



The Range/Output Setup table is the one you will use to set the range limits and assign the discrete outputs. The numbers appearing in it now are the limits and outputs set previously. Note that each box in the table has the three dots (●●●), which indicates that you will need to pick each box, one at a time, in order to set its value.

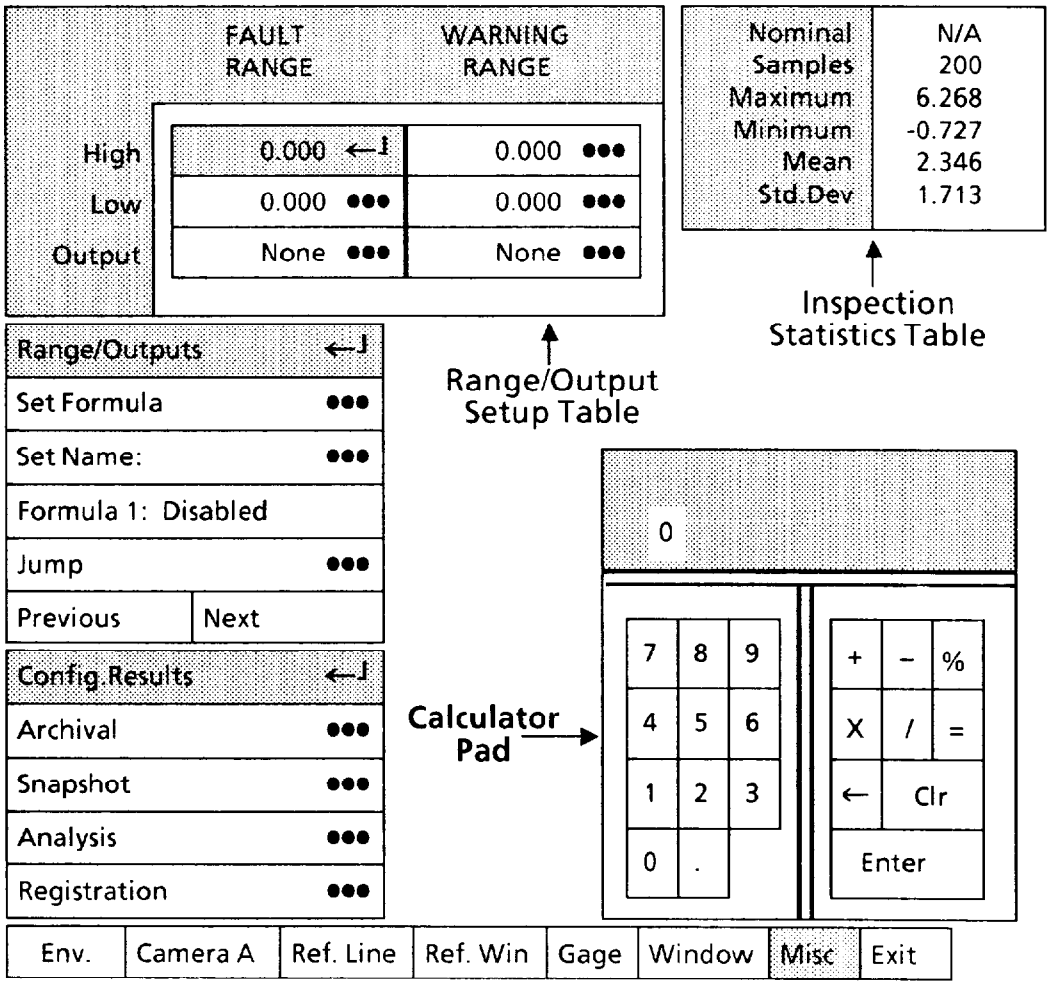
The Inspection Statistics table shows "results" data from a series of trial inspections performed while the CVIM system ran inspections in the "learn" mode. These numbers can help you choose the best values for the range limits. Refer to the *CVIM User's Manual*, Chapter 10, *Runtime Operations*, for more information about this subject.

Assigning Range Limits and Outputs (continued)

The next steps show you how to select values for the range limits.

NOTE: The order in which these steps are presented may not be appropriate in *all* cases. If not, a message will appear that says: VALUE OUT OF RANGE. For example, this message will appear if you try to change the *upper* warning range limit to a value below the *lower* warning range limit.

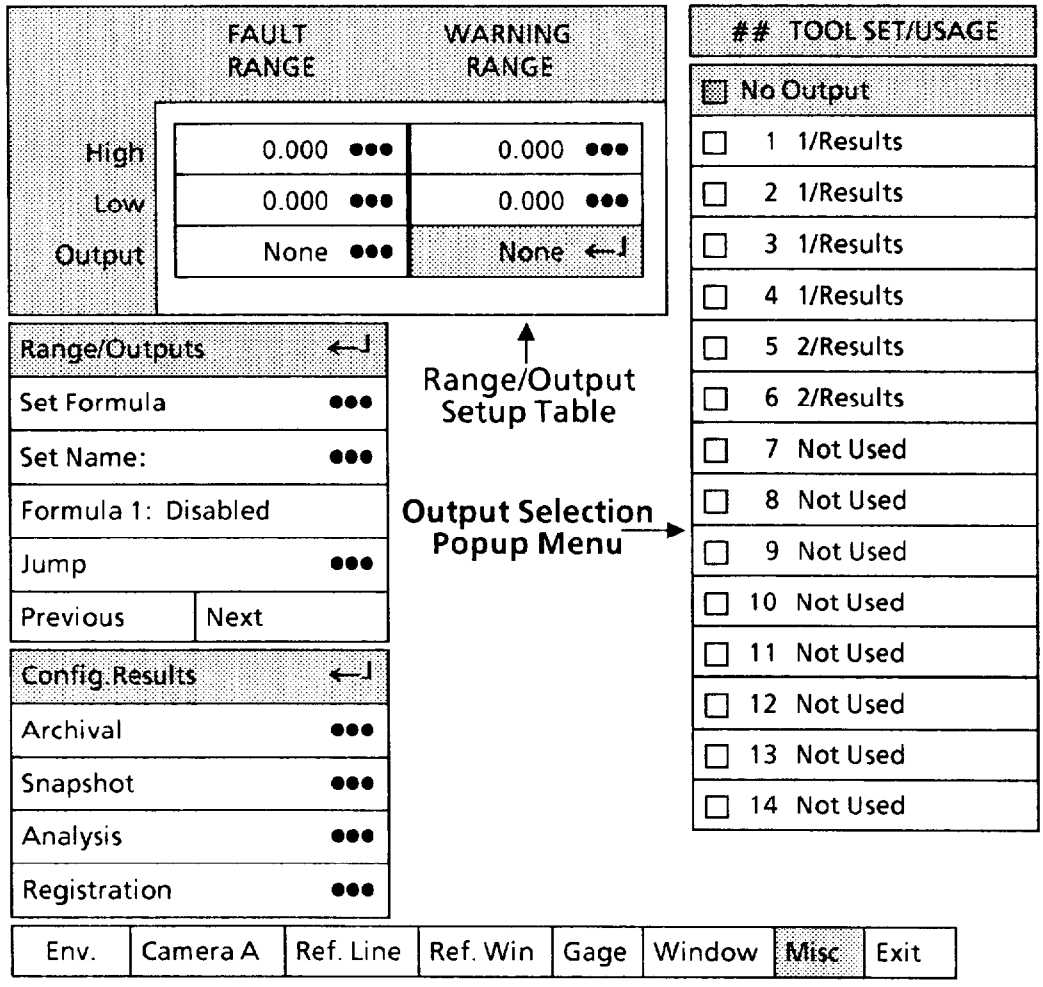
Your Action	Comments
Pick the upper box under FAULT RANGE.	This is the fault range upper ("High") limit. When you pick this box, the calculator pad appears on the screen:



Pick each digit of the upper fault limit value.	As you pick each digit, it appears in the calculator "display." Thus, for a value of 7.5, pick "7," ". (decimal)," and "5."
Pick the Enter *key.*	When you pick the Enter key, the new value will appear in the upper box under FAULT RANGE.
Pick the middle box under FAULT RANGE.	This is the fault range lower ("Low") limit.
Pick each digit of the lower fault limit value.	As you pick each digit, it appears in the calculator "display."

Assigning Range Limits and Outputs (continued)

Your Action	Comments
Pick the Enter *key.*	When you pick the Enter key, the new value will appear in the middle box under FAULT RANGE.
Pick the upper box under WARNING RANGE.	This is the warning range upper ("High") limit.
Pick each digit of the upper warning limit value.	As you pick each digit, it will appear in the calculator "display."
Pick the Enter *key.*	When you pick the Enter key, the new value will appear in the upper box under WARNING RANGE.
Pick the middle box under WARNING RANGE.	This is the warning range lower ("Low") limit.
Pick each digit of the lower warning limit value.	As you pick each digit, it will appear in the calculator "display."
Pick the Enter *key.*	When you pick the Enter key, the new value will appear in the middle box under WARNING RANGE.
Pick the lower box under WARNING RANGE.	When you pick this box, a variation of the Output Assignment popup menu appears on the screen, as follows:



Assigning Range Limits and Outputs (continued)

Your Action	Comments
	This is the Output Selection popup menu. It shows the output functions that you assigned to the Output Assignment popup menu. (Refer to the *CVIM User's Manual*, Chapter 4, *Operating Environment.*)
	NOTE: This menu shows that only the outputs that you designated in Chapter 4 as "1/Results" (or "2/Results" if this formula is in tool set #2) are available to *this* formula. The available outputs appear in light type, and all others appear in **black** type (meaning that you cannot pick them).
	Note also that the No Output box in the Output Selection popup menu has a shaded square (□). This indicates that *no* output is currently assigned to carry WARNING RANGE signals for *this* formula.
Pick the output number for the WARNING RANGE.	From the Output Selection popup menu, pick one of the available output boxes labeled "1/Results" (or "2/Results," if appropriate). When you pick the appropriate box, the shaded square shifts to it.
	In addition, the output number appears in the lower box under WARNING RANGE.
Pick the lower box under FAULT RANGE.	
Pick the output number for the FAULT RANGE.	From the Output Selection popup menu, pick one of the available output boxes labeled "1/Results" (or "2/Results," if appropriate). When you pick the appropriate box, the shaded square will shift to it.
	In addition, the output number appears in the lower box under FAULT RANGE.

Results Data: Size and Rounding Constraints

The limits on the *size* of formula results data – the numbers resulting from a formula's operation – are a function of the memory space allocated to those results.

Here are the specific limits:

- The "largest" *positive* fixed point *result* value that the system can return is 32,767.999.
- The "largest" *negative* fixed point *result* value that the system can return is -32,768.000.
- The largest *positive* integer result from a formula's operation cannot exceed $2^{31} - 1$, or 2,147,483,647.
- The largest *negative* integer result from a formula's operation cannot exceed -2^{31}, or -2,147,483,648.
- Results from fixed point calculations sometimes contain rounding errors. Normally, these errors will not affect the usefulness of the results for your applications.

MATH-PAK Application Examples

This section provides three MATH-PAK application examples, the main purpose of which is to give you some ideas about the types of CVIM inspection application problems that you can solve with MATH-PAK formulas.

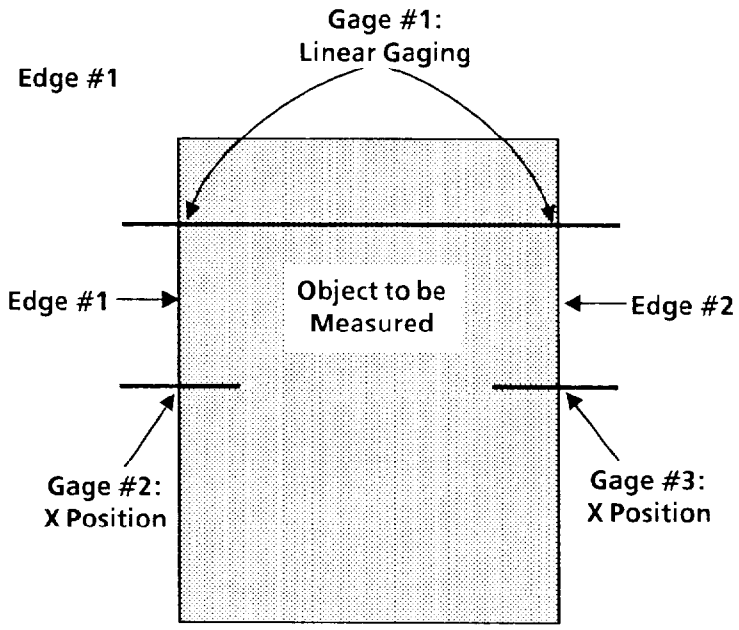
Example #1: Replacing Linear Measurement Gages with Positional Gages Without the MATH-PAK option, an inspection application normally uses a single linear gage to perform a distance measurement.

With the MATH-PAK option, however, some applications could perform the same measurement using two "positional" line gages and a MATH-PAK formula.

A potential benefit of this method is faster measurement, particularly when the gaging mode is gray scale and the measurement crosses most of the image field.

The two-gage measurement can be set up by positioning two linear gages, each configured for either X- or Y-position operation, across the two edges on the object where the measurement is to be made.

This example compares the single-gage method to the two-gage-and-formula method. The gages are positioned across the object as shown in the following illustration:



Note that the measurement edges of the object are parallel to each other.

In the single-gage method, gage #1 measures the distance between edge #1 and #2 of the object. In the two-gage method, gages #2 and #3 return the X positions for edges #1 and #2, respectively. The formula then calculates the distance between edge #1 and #2 by subtracting the gage #2 X position from the gage #3 X position.

MATH-PAK Application Examples (continued)

Here are two example formulas that use the X-position results from gages #2 and #3 to calculate the distance between edges #1 and #2:

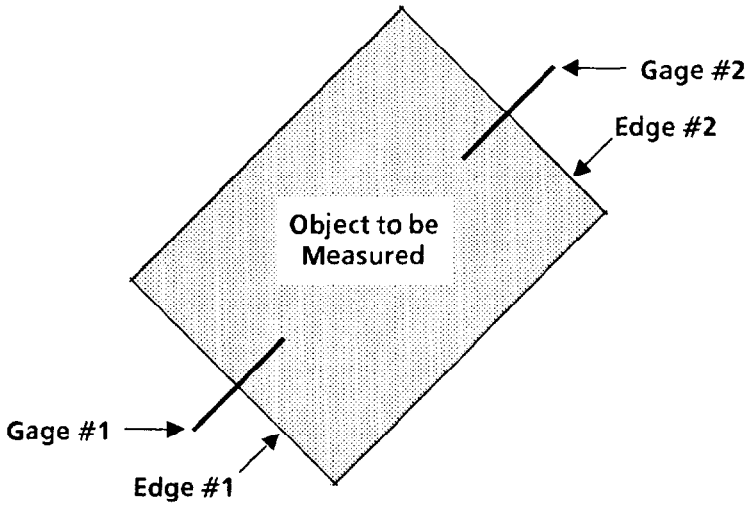
(1) GAGE 3 - GAGE 2

This formula produces a positive result since gage #2 is to the right of gage #1, and the X-position values increase from left to right.(0 to 511 pixels).

(2) ABS(GAGE 2 - GAGE 3)

This formula does not depend on gage #2 being to the right of gage #1. In the expression (GAGE 2 - GAGE 3), the result of the subtraction is a negative number; however, the ABS operator causes the formula to return the *absolute value* of the difference between the two gages.

In some cases, a measurement must be made that is neither horizontal nor vertical, such as measuring the distance between edges #1 and #2 on the following object:



The two "position" gages can still be used, but the formula will be more complex since it must take into account the Y-axis component of the measurement distance in addition to the X-axis component.

Here is an example formula that uses both the X-axis and Y-axis results from gages #1 and #2 to calculate the distance between edges #1 and #2 in the above illustration:

(3) DST(GAGE 1,GAGE 1.3,GAGE2,GAGE2.3)

This formula uses the DST (distance) operator, which calculates the distance between the two edges by using the Pythagorean theorem (refer to the discussion of the DST operator under the *DST Key* heading).

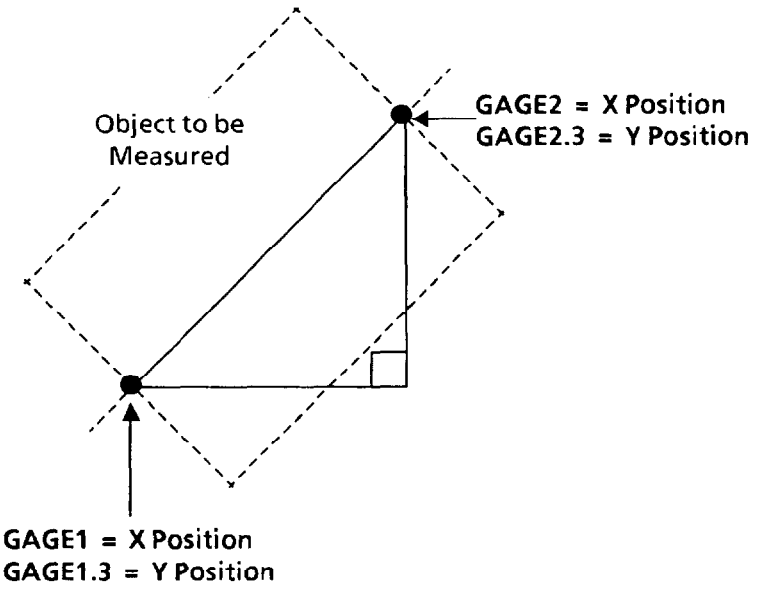
MATH-PAK Application Examples (continued)

In this formula, operands GAGE 1, GAGE1.3, GAGE2, and GAGE2.3 function as follows:

- GAGE 1 returns the X position of the first edge.
- GAGE1.3 returns the Y position of the first edge.
- GAGE 2 returns the X position of the second edge.
- GAGE2.3 returns the Y position of the second edge.

NOTE: The GAGE1.3 and GAGE2.3 operands are functional only during the run mode. During the setup mode, these operands always return a 0 (zero).

The effect of the DST formula is to return the length of the hypotenuse of a right angle triangle. In the present example, this hypotenuse is also the "distance" between two sides of the object:



Note that the DST formula is functionally equivalent to the following formula . . .

SQR(SQ(GAGE 1 - GAGE 2) + SQ(GAGE1.3 - GAGE2.3))

. . . but it is both faster and more compact than the "SQR" formula.

Example #2: Determining Areas of Odd Shapes Using Multiple Widows Without the MATH-PAK option, the CVIM system would normally require using a polygon window to count pixels or objects on (or in) "odd-shaped" objects - those that are not easily covered by a rectangular or elliptical window.

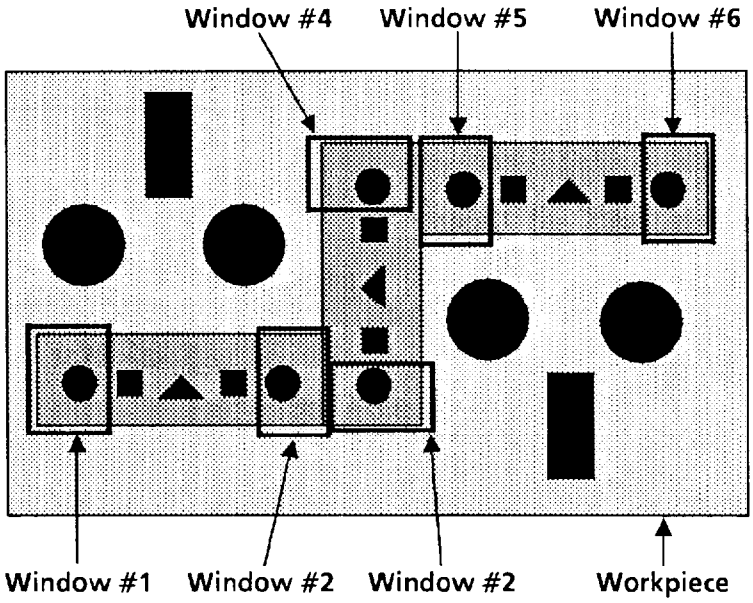
MATH-PAK Application Examples (continued)

With the MATH-PAK option, however, the CVIM system can perform some of these odd-shape applications by using multiple windows and a MATH-PAK formula.

Potential benefits of this method are faster operation and easier setup than polygon windows. Also, this method provides greater flexibility in accommodating changing applications. Thus, by simply adding (or subtracting) individual rectangular windows and changing the formula, you can change the application quickly and easily.

The multiple window measurement can be set up by positioning a window over a workpiece wherever pixel or object counts are needed.

This example uses an odd-shaped part fastened to a workpiece with six rivets. These appear as small, black *circular* objects on the part, as follows:



The objective of this inspection is to determine whether or not the six rivets are in place.

Instead of a single polygon window, six rectangular windows are positioned over the part as shown. Each is configured to count black objects.

The MATH-PAK formula looks like this:

WIN1 + WIN2 + WIN3 + WIN4 + WIN5 + WIN6

The formula sums the black object count from the six windows. Since the correct *count* from each window is *one* (black object), the only correct *sum* is six. Thus, all range limits for the formula would be set to six.

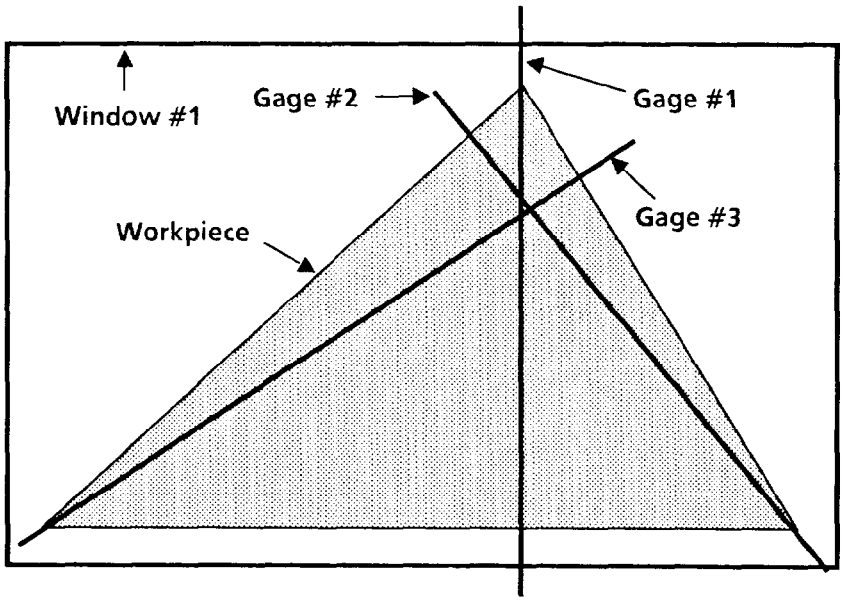
MATH-PAK Application Examples (continued)

Example #3: Making Complex Inspection Decisions Using MATH-PAK Logic Operators Some inspection decisions require more than using inspection results data from tools as the direct basis for making the decision.

With the MATH-PAK option, as shown in Examples #1 and #2, results data from several tools can be combined in a *mathematical* formula, and the results from the formula used, along with range checks, as the basis for a GO-NOGO decision.

Some applications, however, can make use of one or more MATH-PAK *logical* formulas to produce a GO-NOGO decision.

In this example, the requirement is to measure the distance across a trianglular object at three different angles. If two of the three measurements meet or exceed the minimum distance of 15 units, the part's shape is considered to be *uniform*. In addition, a window is used to measure the object's area. If the area meets or exceeds 100 square units, the part's area is considered to be *adequate*.



Two formulas are used, as follows:

(1) (GAGE1 > 15) + (GAGE2 > 15) + (GAGE3 > 15)

(2) (WIN1 > 100) AND (RSLT1 > = 2)

In formula #1, each expression makes the statement, "If GAGEn is greater than 15, then return a logic 1 (*true*); otherwise, return a logic 0 (*false*)."

Formula #1 adds the three expressions; thus, if all three expressions are *true* (logic 1), the formula returns a result of 3. In this example, the result is acceptable if any *two* expressions are true; that is, the formula result is at least 2.

MATH-PAK Application Examples (continued)

In formula #2, the first expression makes the statement, "If WIN1 is greater than 100, then return a logic 1; otherwise, return a logic 0." The second expression makes the statement, "If RSLT1 is greater than or equal to 2, then return a logic 1; otherwise, return a logic 0. Note that RSLT1 is the result from formula #1.

Formula #2 ANDs the expressions (WIN1 > 100) and (RSLT1 > = 2). If *all range limits are set to 1* and both expressions are *true* (logic 1), the formula returns a result of logic 1 (true) and the inspection *passes*; otherwise, the formula returns a logic 0 (false) and the inspection *fails*.

Appendix A CVIM Communications With MATH-PAK

Objectives

This appendix describes:

- CVIM Communications with MATH-PAK.
- Location and format of formula results.
- Formula statistics.
- Setting formula range limits.
- How to access formula results and set configuration.

CVIM With MATH-PAK Communications

The CVIM Communications Manual (Catalog No. 5370-ND002) with Addendum (40062-191-01(A)) describe how to access results data. With MATH-PAK, this information still applies with the following exceptions:

- Results for Windows 1-8 and Gages 1-8 (Block 1)
  are replaced by
  Formulas 1-16
- Results for Windows 9-24 and Gages 9-22 (Block 2)
  are replaced by
  Formulas 17-46
- Results for Gages 23-32 (Block 3)
  are replaced by
  Formulas 47-56
- Fail/Warning Ranges apply to formulas.
- Warning and Fault Flags apply to formulas.

Note: The formula results only replace window or gage results if the formula is enabled. If you want to obtain window or gage results that are replaced by formula results, use the configurable results block (block #4).

Location of Formula Results

When enabled, the formula results replace window and gage results.

- Table A.1 shows the location of the Block 1 formula results.
- Table A.2 shows the location of the Block 2 formula results.
- Table A.3 shows the location of the Block 3 formula results.

Note: You cannot program the programmable results block (block #4) to contain formula results.

Location of Formula Results *(continued)*

Table A.1
Block #1 Formula Results

Word #			Bit #	Definition With MATH-PAK (without MATH-PAK)	Usage	Notes
Backplane		RS-232 and Remote I/O				
Toolset 1	Toolset 2					
48-49	312-313	24-25	0-15	Formula #1 Results (Window #1 Value)		
50-51	314-315	26-27	0-15	Formula #2 Results (Window #2 Value)		
52-53	316-317	28-29	0-15	Formula #3 Results (Window #3 Value)		
54-55	318-319	30-31	0-15	Formula #4 Results (Window #4 Value)		
56-57	320-321	32-33	0-15	Formula #5 Results (Window #5 Value)		
58-59	322-323	34-35	0-15	Formula #6 Results (Window #6 Value)		
60-61	324-325	36-37	0-15	Formula #7 Results (Window #7 Value)		
62-63	326-327	38-39	0-15	Formula #8 Results (Window #8 Value)		
64-65	328-329	40-41	0-15	Formula #9 Results (Gage #1 Value)		
66-67	330-331	42-43	0-15	Formula #10Results (Gage #2 Value)		
68-69	332-333	44-45	0-15	Formula #11 Results (Gage #3 Value)		
70-71	334-335	46-47	0-15	Formula #12 Results (Gage #4 Value)		
72-73	336-337	48-49	0-15	Formula #13 Results (Gage #5 Value)		
74-75	338-339	50-51	0-15	Formula #14 Results (Gage #6 Value)		
76-77	340-341	52-53	0-15	Formula #15 Results (Gage #7 Value)		
78-79	342-343	54-55	0-15	Formula #16 Results (Gage #8 Value)		

Location of Formula Results (continued)

Table A.2
Block 2 Formula Results

Word #			Bit #	Definition With MATH-PAK (without MATH-PAK)	Usage	Notes
Backplane		RS-232 and Remote I/O				
Toolset 1	Toolset 2					
90-91	354-355	2-3	0-15	Formula #17 Results (Window #9 Value)		
92-93	356-357	4-5	0-15	Formula #18 Results (Window #10 Value)		
94-95	358-359	6-7	0-15	Formula #19 Results (Window #11 Value)		
96-97	360-361	8-9	0-15	Formula #20 Results (Window #12 Value)		
98-99	362-363	10-11	0-15	Formula #21 Results (Window #13 Value)		
100-101	364-365	12-13	0-15	Formula #22 Results (Window #14 Value)		
102-103	366-367	14-15	0-15	Formula #23 Results (Window #15 Value)		
104-105	368-369	16-17	0-15	Formula #24 Results (Window #16 Value)		
106-107	370-371	18-19	0-15	Formula #25 Results (Gage #9 Value)		
108-109	372-373	20-21	0-15	Formula #26 Results (Gage #10 Value)		
110-111	374-375	22-23	0-15	Formula #27 Results (Gage #11 Value)		
112-113	376-377	24-25	0-15	Formula #28 Results (Gage #12 Value)		
114-115	378-379	26-27	0-15	Formula #29 Results (Gage #13 Value)		
116-117	380-381	28-29	0-15	Formula #30 Results (Gage #14 Value)		
118-119	382-383	30-31	0-15	Formula #31 Results (Gage #15 Value)		

Location of Formula Results *(continued)*

Table A.2
Block 2 Formula Results (Continued)

Word #			Bit #	Definition With MATH-PAK (without MATH-PAK)	Usage	Notes
Backplane		RS-232 and Remote I/O				
Toolset 1	Toolset 2					
120-121	384-385	32-33	0-15	Formula #32 Results (Gage #16 Value)		
122-123	386-387	34-35	0-15	Formula #33 Results (Window #17 Value)		
124-125	388-389	36-37	0-15	Formula #34 Results (Window #18 Value)		
126-127	390-391	38-39	0-15	Formula #35 Results (Window #19 Value)		
128-129	392-393	40-41	0-15	Formula #36 Results (Window #20 Value)		
130-131	394-395	42-43	0-15	Formula #37 Results (Window #21 Value)		
132-133	396-397	44-45	0-15	Formula #38 Results (Window #22 Value)		
134-135	398-399	46-47	0-15	Formula #39 Results (Window #23 Value)		
136-137	400-401	48-49	0-15	Formula #40 Results (Window #24 Value)		
138-139	402-403	50-51	0-15	Formula #41 Results (Gage #17 Value)		
140-141	404-405	52-53	0-15	Formula #42 Results (Gage #18 Value)		
142-143	406-407	54-55	0-15	Formula #43 Results (Gage #19 Value)		
144-145	408-409	56-57	0-15	Formula #44 Results (Gage #20 Value)		
146-147	410-411	58-59	0-15	Formula #45 Results (Gage #21 Value)		
148-149	412-413	60-61	0-15	Formula #46 Results (Gage #22 Value)		

Location of Formula Results (continued)

Table A.3
Block 3 Formula Results

Word #			Bit #	Definition With MATH-PAK (without MATH-PAK)	Usage	Notes
Backplane		RS-232 and Remote I/O				
Toolset 1	Toolset 2					
154-155	418-419	2-3	0-15	Formula #47 Results (Gage #23 Value)		
156-157	420-421	4-5	0-15	Formula #48 Results (Gage #24 Value)		
158-159	422-423	6-7	0-15	Formula #49 Results (Gage #25 Value)		
160-161	424-425	8-9	0-15	Formula #50 Results (Gage #26 Value)		
162-163	426-427	10-11	0-15	Formula #51 Results (Gage #27 Value)		
164-165	428-429	12-13	0-15	Formula #52 Results (Gage #28 Value)		
166-167	430-431	14-15	0-15	Formula #53 Results (Gage #29 Value)		
168-169	432-433	16-17	0-15	Formula #54 Results (Gage #30 Value)		
170-171	434-435	18-19	0-15	Formula #55 Results (Gage #31 Value)		
172-173	436-437	20-21	0-15	Formula #56 Results (Gage #32 Value)		

Data Formats For Formula Results

The data type for each formula is determined by the equation:

- If the formula contains *any* items that are fixed point values, the formula results will be a fixed point value (16.16).
- If the formula *only* contains integer values, the formula results will be an integer value (32.0)

Obtaining Formula Results and Statistics

Obtain formula results and statistics the same way you would toolset results. Refer to Publication 5370-ND002 and Addendum (40062-191-01(A)).

Statistics Block

When using the statistics block and formulas are enabled, please note that an enabled formula takes the place of the corresponding tool in the statistics block.

For example: If you request statistics for Gage #1 and formula #9 is enabled, the statistics for the formula will appear in the statistics block.

Configuration Blocks

When the corresponding formula is enabled, the configuration block fail and warning ranges are applied to the formula results.

- Table A.4 lists Fail/Warning Range (Toolset 1) data locations for Formulas 9-16, 25-32, and 41-56 (Configuration blocks 10-41).
- Table A.5 lists Fail/Warning Range (Toolset 1) data locations for Formulas 1-8, 17-24, and 33-40 (Configuration blocks 42-65).
- Table A.6 lists Fail/Warning Range (Toolset 2) data locations for Formulas 9-16, 25-32, and 41-56 (Configuration blocks 72-103).
- Table A.7 lists Fail/Warning Range (Toolset 2) data locations for Formulas 9-16, 25-32, and 41-56 (Configuration blocks 104-127).

Fail/Warning Ranges: Formulas # 9-16, 25-32, 41-56 in Toolset 1

Table A.4 lists the fail and warning range data locations for Formulas 9 through 16, 25 through 32, and 41 through 56 (blocks 10-41). Refer to the following for the block number of each formula. Table A.4 provides the word # of the data in each block.

Formula 9- Block 10
Formula 10- Block 11
Formula 11- Block 12
Formula 12- Block 13
Formula 13- Block 14
Formula 14- Block 15
Formula 15- Block 16
Formula 16- Block 17
Formula 25- Block 18
Formula 26- Block 19
Formula 27- Block 20
Formula 28- Block 21
Formula 29- Block 22
Formula 30- Block 23
Formula 31- Block 24
Formula 32- Block 25
Formula 41- Block 26
Formula 42- Block 27
Formula 43- Block 28
Formula 44- Block 29
Formula 45- Block 30
Formula 46- Block 31
Formula 47- Block 32
Formula 48- Block 33
Formula 49- Block 34
Formula 50- Block 35
Formula 51- Block 36
Formula 52- Block 37
Formula 53- Block 38
Formula 54- Block 39
Formula 55- Block 40
Formula 56- Block 41

Table A.4
Configuration Blocks #10-41 Fail/Warning Ranges (TS1)

Remote I/O & RS-232 Word #*	Bit #	Definition	Usage	Notes
1	14	Formula Enable	1 = Enable	
17	0-15	Fail Range High (Integer)		Words 17 and 18 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
18	0-15	Fail Range High (Fraction)		
19	0-15	Fail Range Low (Integer)		Words 19 and 20 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
20	0-15	Fail Range Low (Fraction)		
21	0-15	Warning Range High (Integer)		Words 21 and 22 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
22	0-15	Warning Range High (Fraction)		
23	0-15	Warning Range Low (Integer)		Words 23 and 24 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
24	0-15	Warning Range Low (Fraction)		

Fail/Warning Ranges: Formulas # 1-8, 17-24, 33-40 in Toolset 1

Table A.5 lists the fail and warning range data locations for Formulas 1 through 8, 17 through 24, and 33 through 40 (blocks 42-65). Refer to the following for the block number for each formula. Table A.5 provides the word # of the data in each block.

Formula 1- Block 42
Formula 2- Block 43
Formula 3- Block 44
Formula 4- Block 45
Formula 5- Block 46
Formula 6- Block 47
Formula 7- Block 48
Formula 8- Block 49
Formula 17- Block 50
Formula 18- Block 51
Formula 19- Block 52
Formula 20- Block 53
Formula 21- Block 54
Formula 22- Block 55
Formula 23- Block 56
Formula 24- Block 57
Formula 33- Block 58
Formula 34- Block 59
Formula 35- Block 60
Formula 36- Block 61
Formula 37- Block 62
Formula 38- Block 63
Formula 39- Block 64
Formula 40- Block 65

Table A.5
Configuration Blocks #42-65 Fail/Warning Ranges (TS1)

Remote I/O & RS-232 Word #*	Bit #	Definition	Usage	Notes
1	14	Formula Enable	1 = Enable	
28	0-15	Fail Range High (Integer)		Words 28 and 29 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
29	0-15	Fail Range High (Fraction)		
30	0-15	Fail Range Low (Integer)		Words 30 and 31 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
31	0-15	Fail Range Low (Fraction)		
32	0-15	Warning Range High (Integer		Words 32 and 33 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
33	0-15	Warning Range High (Fraction		
34	0-15	Warning Range Low (Integer)		Words 34 and 35 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
35	0-15	Warning Range Low (Fraction)		

Fail/Warning Ranges: Formulas # 9-16, 25-32, 41-56 in Toolset 2

Table A.6 lists the fail and warning range data locations for Formulas 9 through 16, 25 through 32, and 41 through 56 (blocks 72-103). Refer to the following for the block number for each formula. Table A.6 provides the word # of the data in each block.

Formula 9- Block 72
Formula 10- Block 73
Formula 11- Block 74
Formula 12- Block 75
Formula 13- Block 76
Formula 14- Block 77
Formula 15- Block 78
Formula 16- Block 79
Formula 25- Block 80
Formula 26- Block 81
Formula 27- Block 82
Formula 28- Block 83
Formula 29- Block 84
Formula 30- Block 85
Formula 31- Block 86
Formula 32- Block 87
Formula 41- Block 88
Formula 42- Block 89
Formula 43- Block 90
Formula 44- Block 91
Formula 45- Block 92
Formula 46- Block 93
Formula 47- Block 94
Formula 48- Block 95
Formula 49- Block 96
Formula 50- Block 97
Formula 51- Block 98
Formula 52- Block 99
Formula 53- Block 100
Formula 54- Block 101
Formula 55- Block 102
Formula 56- Block 103

Table A.6
Configuration Blocks #72-103 Fail/Warning Ranges (TS2)

Remote I/O & RS-232 Word #*	Bit #	Definition	Usage	Notes
1	14	Formula Enable	1 = Enable	
17	0-15	Fail Range High (Integer)		Words 17 and 18 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
18	0-15	Fail Range High (Fraction)		
19	0-15	Fail Range Low (Integer)		Words 19 and 20 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
20	0-15	Fail Range Low (Fraction)		
21	0-15	Warning Range High (Integer)		Words 21 and 22 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
22	0-15	Warning Range High (Fraction)		
23	0-15	Warning Range Low (Integer)		Words 23 and 24 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
24	0-15	Warning Range Low (Fraction)		

Fail/Warning Ranges: Formulas # 1-8, 17-24, 33-40 in Toolset 2

Table A.7 lists the fail and warning range data locations for Formulas 1 through 8, 17 through 24, and 33 through 40 (blocks 104-127). Refer to the following for the block number for each formula. Table A.7 provides the word # of the data in each block.

Formula 1- Block 104
Formula 2- Block 105
Formula 3- Block 106
Formula 4- Block 107
Formula 5- Block 108
Formula 6- Block 109
Formula 7- Block 110
Formula 8- Block 111
Formula 17- Block 112
Formula 18- Block 113
Formula 19- Block 114
Formula 20- Block 115
Formula 21- Block 116
Formula 22- Block 117
Formula 23- Block 118
Formula 24- Block 119
Formula 33- Block 120
Formula 34- Block 121
Formula 35- Block 122
Formula 36- Block 123
Formula 37- Block 124
Formula 38- Block 125
Formula 39- Block 126
Formula 40- Block 127

Table A.7
Configuration Blocks #104-127 Fail/Warning Ranges (TS2)

Remote I/O & RS-232 Word #*	Bit #	Definition	Usage	Notes
1	14	Formula Enable	1 = Enable	
28	0-15	Fail Range High (Integer)		Words 28 and 29 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
29	0-15	Fail Range High (Fraction)		
30	0-15	Fail Range Low (Integer)		Words 30 and 31 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
31	0-15	Fail Range Low (Fraction)		
32	0-15	Warning Range High (Integer		Words 32 and 33 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
33	0-15	Warning Range High (Fraction		
34	0-15	Warning Range Low (Integer)		Words 34 and 35 represent a 16 (bit).16 (bit) fixed point decimal value or 32 bit integer.
35	0-15	Warning Range Low (Fraction)		

Discrete Bit Inputs

When formulas are enabled, the Warning and Fault flags apply to the corresponding formula. The formula bits only take the place of the toolset bits if a formula is enabled. Table A.8 lists the location of the formula warning and fault bits.

Table A.8
Formula Discrete Bit Inputs

Word #			Bit#		Definition	Usage	Notes
Backplane		RS-232 and Remote I/O					
Toolset 1	Toolset 2		PI	PLC			
17	281	1	0	0	Formula 1 Warning Flag	0 = Pass 1 = Fail	
17	281	1	1	1	Formula 1 Fault Flag	0 = Pass 1 = Fail	
17	281	1	2	2	Formula 2 Warning Flag	0 = Pass 1 = Fail	
17	281	1	3	3	Formula 2 Fault Flag	0 = Pass 1 = Fail	
17	281	1	4	4	Formula 3 Warning Flag	0 = Pass 1 = Fail	
17	281	1	5	5	Formula 3 Fault Flag	0 = Pass 1 = Fail	
17	281	1	6	6	Formula 4 Warning Flag	0 = Pass 1 = Fail	
17	281	1	7	7	Formula 4 Fault Flag	0 = Pass 1 = Fail	
17	281	1	8	10	Formula 5 Warning Flag	0 = Pass 1 = Fail	

Discrete Bit Inputs
(continued)

Table A.8
Discrete Bit Inputs Continued

Word #			Bit #		Definition	Usage	Notes
Backplane		RS-232 and Remote I/O					
Toolset 1	Toolset 2		PI	PLC			
17	281	1	9	11	Formula 5 Fault Flag	0 = Pass 1 = Fail	
17	281	1	10	12	Formula 6 Warning Flag	0 = Pass 1 = Fail	
17	281	1	11	13	Formula 6 Fault Flag	0 = Pass 1 = Fail	
17	281	1	12	14	Formula 7 Warning Flag	0 = Pass 1 = Fail	
17	281	1	13	15	Formula 7 Fault Flag	0 = Pass 1 = Fail	
17	281	1	14	16	Formula 8 Warning Flag	0 = Pass 1 = Fail	
17	281	1	15	17	Formula 8 Fault Flag	0 = Pass 1 = Fail	
18	282	2	0	0	Formula 17 Warning Flag	0 = Pass 1 = Fail	
18	282	2	1	1	Formula 17 Fault Flag	0 = Pass 1 = Fail	
18	282	2	2	2	Formula 18 Warning Flag	0 = Pass 1 = Fail	
18	282	2	3	3	Formula 18 Fault Flag	0 = Pass 1 = Fail	
18	282	2	4	4	Formula 19 Warning Flag	0 = Pass 1 = Fail	
18	282	2	5	5	Formula 19 Fault Flag	0 = Pass 1 = Fail	
18	282	2	6	6	Formula 20 Warning Flag	0 = Pass 1 = Fail	
18	282	2	7	7	Formula 20 Fault Flag	0 = Pass 1 = Fail	

Discrete Bit Inputs
(continued)

Table A.8
Discrete Bit Inputs Continued

Word #			Bit #		Definition	Usage	Notes
Backplane		RS-232 and Remote I/O					
Toolset 1	Toolset 2		PI	PLC			
18	282	2	8	10	Formula 21 Warning Flag	0 = Pass 1 = Fail	
18	282	2	9	11	Formula 21 Fault Flag	0 = Pass 1 = Fail	
18	282	2	10	12	Formula 22 Warning Flag	0 = Pass 1 = Fail	
18	282	2	11	13	Formula 22 Fault Flag	0 = Pass 1 = Fail	
18	282	2	12	14	Formula 23 Warning Flag	0 = Pass 1 = Fail	
18	282	2	13	15	Formula 23 Fault Flag	0 = Pass 1 = Fail	
18	282	2	14	16	Formula 24 Warning Flag	0 = Pass 1 = Fail	
18	282	2	15	17	Formula 24 Fault Flag	0 = Pass 1 = Fail	
19	283	3	0	0	Formula 33 Warning Flag	0 = Pass 1 = Fail	
19	283	3	1	1	Formula 33 Fault Flag	0 = Pass 1 = Fail	
19	283	3	2	2	Formula 34 Warning Flag	0 = Pass 1 = Fail	
19	283	3	3	3	Formula 34 Fault Flag	0 = Pass 1 = Fail	
19	283	3	4	4	Formula 35 Warning Flag	0 = Pass 1 = Fail	
19	283	3	5	5	Formula 35 Fault Flag	0 = Pass 1 = Fail	
19	283	3	6	6	Formula 36 Warning Flag	0 = Pass 1 = Fail	

Discrete Bit Inputs
(continued)

Table A.8
Discrete Bit Inputs Continued

Word #			Bit #		Definition	Usage	Notes
Backplane		RS-232 and Remote I/O					
Toolset 1	Toolset 2		PI	PLC			
19	283	3	7	7	Formula 36 Fault Flag	0 = Pass 1 = Fail	
19	283	3	8	10	Formula 37 Warning Flag	0 = Pass 1 = Fail	
19	283	3	9	11	Formula 37 Fault Flag	0 = Pass 1 = Fail	
19	283	3	10	12	Formula 38 Warning Flag	0 = Pass 1 = Fail	
19	283	3	11	13	Formula 38 Fault Flag	0 = Pass 1 = Fail	
19	283	3	12	14	Formula 39 Warning Flag	0 = Pass 1 = Fail	
19	283	3	13	15	Formula 39 Fault Flag	0 = Pass 1 = Fail	
19	283	3	14	16	Formula 40 Warning Flag	0 = Pass 1 = Fail	
19	283	3	15	17	Formula 40 Fault Flag	0 = Pass 1 = Fail	
20	284	4	0	0	Formula 9 Warning Flag	0 = Pass 1 = Fail	
20	284	4	1	1	Formula 9 Fault Flag	0 = Pass 1 = Fail	
20	284	4	2	2	Formula 10 Warning Flag	0 = Pass 1 = Fail	
20	284	4	3	3	Formula 10 Fault Flag	0 = Pass 1 = Fail	

Discrete Bit Inputs
(continued)

Table A.8
Discrete Bit Inputs Continued

Word #			Bit #		Definition	Usage	Notes
Backplane		RS-232 and Remote I/O					
Toolset 1	Toolset 2		PI	PLC			
20	284	4	4	4	Formula 11 Warning Flag	0 = Pass 1 = Fail	
20	284	4	5	5	Formula 11 Fault Flag	0 = Pass 1 = Fail	
20	284	4	6	6	Formula 12 Warning Flag	0 = Pass 1 = Fail	
20	284	4	7	7	Formula 12 Fault Flag	0 = Pass 1 = Fail	
20	284	4	8	10	Formula 13 Warning Flag	0 = Pass 1 = Fail	
20	284	4	9	11	Formula 13 Fault Flag	0 = Pass 1 = Fail	
20	284	4	10	12	Formula 14 Warning Flag	0 = Pass 1 = Fail	
20	284	4	11	13	Formula 14 Fault Flag	0 = Pass 1 = Fail	
20	284	4	12	14	Formula 15 Warning Flag	0 = Pass 1 = Fail	
20	284	4	13	15	Formula 15 Fault Flag	0 = Pass 1 = Fail	
20	284	4	14	16	Formula 16 Warning Flag	0 = Pass 1 = Fail	
20	284	4	15	17	Formula 16 Fault Flag	0 = Pass 1 = Fail	
21	285	5	0	0	Formula 25 Warning Flag	0 = Pass 1 = Fail	
21	285	5	1	1	Formula 25 Fault Flag	0 = Pass 1 = Fail	

Discrete Bit Inputs
(continued)

Table A.8
Discrete Bit Inputs Continued

Word #			Bit #		Definition	Usage	Notes
Backplane		RS-232 and Remote I/O					
Toolset 1	Toolset 2		PI	PLC			
21	285	5	2	2	Formula 26 Warning Flag	0 = Pass 1 = Fail	
21	285	5	3	3	Formula 26 Fault Flag	0 = Pass 1 = Fail	
21	285	5	4	4	Formula 27 Warning Flag	0 = Pass 1 = Fail	
21	285	5	5	5	Formula 27 Fault Flag	0 = Pass 1 = Fail	
21	285	5	6	6	Formula 28 Warning Flag	0 = Pass 1 = Fail	
21	285	5	7	7	Formula 28 Fault Flag	0 = Pass 1 = Fail	
21	285	5	8	10	Formula 29 Warning Flag	0 = Pass 1 = Fail	
21	285	5	9	11	Formula 29 Fault Flag	0 = Pass 1 = Fail	
21	285	5	10	12	Formula 30 Warning Flag	0 = Pass 1 = Fail	
21	285	5	11	13	Formula 30 Fault Flag	0 = Pass 1 = Fail	
21	285	5	12	14	Formula 31 Warning Flag	0 = Pass 1 = Fail	
21	285	5	13	15	Formula 31 Fault Flag	0 = Pass 1 = Fail	
21	285	5	14	16	Formula 32 Warning Flag	0 = Pass 1 = Fail	
21	285	5	15	17	Formula 32 Fault Flag	0 = Pass 1 = Fail	

Discrete Bit Inputs
(continued)

Table A.8
Discrete Bit Inputs Continued

Word #			Bit #		Definition	Usage	Notes
Backplane		RS-232 and Remote I/O					
Toolset 1	Toolset 2		PI	PLC			
22	286	6	0	0	Formula 41 Warning Flag	0 = Pass 1 = Fail	
22	286	6	1	1	Formula 41 Fault Flag	0 = Pass 1 = Fail	
22	286	6	2	2	Formula 42 Warning Flag	0 = Pass 1 = Fail	
22	286	6	3	3	Formula 42 Fault Flag	0 = Pass 1 = Fail	
22	286	6	4	4	Formula 43 Warning Flag	0 = Pass 1 = Fail	
22	286	6	5	5	Formula 43 Fault Flag	0 = Pass 1 = Fail	
22	286	6	6	6	Formula 44 Warning Flag	0 = Pass 1 = Fail	
22	286	6	7	7	Formula 44 Fault Flag	0 = Pass 1 = Fail	
22	286	6	8	10	Formula 45 Warning Flag	0 = Pass 1 = Fail	
22	286	6	9	11	Formula 45 Fault Flag	0 = Pass 1 = Fail	
22	286	6	10	12	Formula 46 Warning Flag	0 = Pass 1 = Fail	
22	286	6	11	13	Formula 46 Fault Flag	0 = Pass 1 = Fail	
22	286	6	12	14	Formula 47 Warning Flag	0 = Pass 1 = Fail	
22	286	6	13	15	Formula 47 Fault Flag	0 = Pass 1 = Fail	

Discrete Bit Inputs
(continued)

Table A.8
Discrete Bit Inputs Continued

Word #			Bit #		Definition	Usage	Notes
Backplane		RS-232 and Remote I/O					
Toolset 1	Toolset 2		PI	PLC			
22	286	6	14	16	Formula 48 Warning Flag	0 = Pass 1 = Fail	
22	286	6	15	17	Formula 48 Fault Flag	0 = Pass 1 = Fail	
23	287	7	0	0	Formula 49 Warning Flag	0 = Pass 1 = Fail	
23	287	7	1	1	Formula 49 Fault Flag	0 = Pass 1 = Fail	
23	287	7	2	2	Formula 50 Warning Flag	0 = Pass 1 = Fail	
23	287	7	3	3	Formula 50 Fault Flag	0 = Pass 1 = Fail	
23	287	7	4	4	Formula 51 Warning Flag	0 = Pass 1 = Fail	
23	287	7	5	5	Formula 51 Fault Flag	0 = Pass 1 = Fail	
23	287	7	6	6	Formula 52 Warning Flag	0 = Pass 1 = Fail	
23	287	7	7	7	Formula 52 Fault Flag	0 = Pass 1 = Fail	
23	287	7	8	10	Formula 53 Warning Flag	0 = Pass 1 = Fail	
23	287	7	9	11	Formula 53 Fault Flag	0 = Pass 1 = Fail	
23	287	7	10	12	Formula 54 Warning Flag	0 = Pass 1 = Fail	
23	287	7	11	13	Formula 54 Fault Flag	0 = Pass 1 = Fail	

Discrete Bit Inputs
(continued)

Table A.8
Discrete Bit Inputs Continued

Word #			Bit #		Definition	Usage	Notes
Backplane		RS-232 and Remote I/O					
Toolset 1	Toolset 2		PI	PLC			
23	287	7	12	14	Formula 55 Warning Flag	0 = Pass 1 = Fail	
23	287	7	13	15	Formula 55 Fault Flag	0 = Pass 1 = Fail	
23	287	7	14	16	Formula 56 Warning Flag	0 = Pass 1 = Fail	
23	287	7	15	17	Formula 56 Fault Flag	0 = Pass 1 = Fail	

ALLEN-BRADLEY
A ROCKWELL INTERNATIONAL COMPANY

A subsidiary of Rockwell International, one of the world's largest technology companies, Allen-Bradley meets today's automation challenges with over 85 years of practical plant floor experience. More than 13,000 employees throughout the world design, manufacture and apply a wide range of control and automation products and supporting services to help our customers continuously improve quality, productivity and time to market. These products and services not only control individual machines, but also integrate the manufacturing process while providing access to vital plant floor data that can be used to support decision-making throughout the enterprise.

With offices in major cities worldwide.

WORLD HEADQUARTERS
1201 South Second Street
Milwaukee, WI 53204 USA
Tel: (414)382-2000
Telex: 43 11 016
FAX: (414)382-4444

EUROPE/MIDDLE EAST/ AFRICA HEADQUARTERS
Allen-Bradley Europe B.V.
Amsterdamseweg 15
1422 AC Uithoorn
The Netherlands
Tel: (31)2975/43500
Telex: (844) 18042
FAX: (31)2975/60222

ASIA/PACIFIC HEADQUARTERS
Allen-Bradley (Hong Kong) Limited
Room 1006, Block B, Sea View Estate
2-8 Watson Road
Hong Kong
Tel: (852)887-4788
Telex: (780) 64347
FAX: (852)510-9436

CANADA HEADQUARTERS
Allen-Bradley Canada Limited
135 Dundas Street
Cambridge, Ontario N1R 5X1
Canada
Tel: (519)623-1810
FAX: (519)623-8930

LATIN AMERICA HEADQUARTERS
1201 South Second Street
Milwaukee, WI 53204 USA
Tel: (414)382-2000
Telex: 43 11 016
FAX: (414)382-2400

Catalog No. 5370-ND005 Series A

40062-194-01(A)